SEMESTER -3

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT201	DATA STRUCTURES	PCC	3	1	0	4

Preamble: The syllabus is prepared with the view of preparing the Engineering Graduates capable of understanding essential concept of data structures, designing algorithms to perform operations involving these data structures and to choose appropriate data structures to solve real world problems.

Prerequisite: programming in C

Course Outcomes: After the completion of the course the student will be able to

CO_No.	Course Outcome(CO)	Bloom's Category
CO 1	**Summarize** different categories of data structures	Level 2 : Understand
CO 2	**Identify different** parameters to analyze the performance of an algorithm.	Level 3 : Apply
CO 3	**Explain** the significance of dynamic memory management Techniques.	Level 2 : Understand
CO 4	**Design** algorithms to perform operations with Linear and Nonlinear data structures	Level 3 : Apply
CO 5	**Illustrate** various technique to for searching, Sorting and hashing	Level 2 : Understand
CO 6	**Choose** appropriate data structures to solve real world problems efficiently.	Level 3 : Apply

Mapping of course outcomes with program outcomes

COs	PROGRAMME OUTCOMES (PO)											
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	2	-	-	1	-	-	-	-	-	1
CO 2	3	2	2	2	1	1	-	-	-	-	-	1
CO 3	3	3	3	2	1	1	-	-	-	-	-	1
CO 4	3	3	3	2	1	1	-	-	-	-	-	1
CO 5	3	2	2	1	1	-	-	-	-	-	-	1
CO 6	3	3	3	2	1	1	-	-	-	-	-	1

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	20	20
Apply	20	20	70
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Test (2 numbers) : 25 marks

Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Differentiate CDT and ADT
2. Classify classical datstructures
3. Compare array and linked list
4. Represent single and double dimensional array
5. Describe any three applications of array

Course Outcome 2 (CO2):

1. Identify the needs of algorithm analysis
2. Select two parameters to do the performance analysis of an algorithm
3. Identify 3 possible cases of time complexity

Course Outcome 3 (CO3):

1. Classify linked list
2. Illustrate different operations on singly, doubly and circular linked list
3. Represent linked list in memory (static and dynamic).

4. Summarize different dynamic memory management schemes.
5. Demonstrate the first fit , best fit , worst fit and next fit allocation of given process queue and free list

Process queue

85 K	35 K	70 K	100 K

Free list

80K	130K	90 K	40 K

Course Outcome 4 (CO4):

1. Design the algorithms to perform PUSH()and POP() and STATUS() operations on stack using array and linked list
2. Apply Stack data structure in infix to postfix conversion, expression evaluation, recursion and delimiter matching.
3. Design the algorithms for pre-order, in-order and post-order traversal on binary trees
4. Develop the algorithms for ENQUEUE(), and DEQUEUE() operations on queue data structures
5. Construct the algorithms for graph traversal(BFS, DFS)

Course Outcome 5 (CO5):

1. Classify Sorting Techniques (internal and external, n2 and n logn)
2. Compare Linear and binary search
3. Illustrate bubble, selection and insertion sort.
4. Describe quick and merge sort
5. Represent the following values in the given order in a hash table (Size of hash table is 7 and hash function used is h(k)=k mod 7) for each of the scenario.
   19, 26, 13, 48, 17
   a) When collisions are handled by linear probing
   b) When collisions are handled by double using second hash function h=5-(k mod 5)

Course Outcome 6 (CO6):

1. Develop an application program which is to be used in Ticket counter, where First person gets ticket first and go out first, using suitable data structure.
2. Make use of suitable data structure to store the details of pass percentage of the college in chronological order of years (oldest to newest) and retrieve the information in reverse chronological order of years, using suitable data structure.

Model Question Paper

PART A **(10*3=30)**

(Each Question carries 3 Marks)

1. Classify Sorting Techniques
2. Differentiate CDT and ADT
3. Classify linked list
4. Compare First fit and Next Fit Algorithms
5. Design the algorithms to perform PUSH()and POP()
6. Summarize operations Queue data structure
7. List out the features of binary tree
8. Explain binary sorting
9. Define hashing
10. Illustrate separate chaining with an example

PART B **(5*14=70)**

11. Classify classical data structures

OR

12. Illustrate Quick sort with the help of an example

13. Illustrate different operations on singly linked list

OR

14. Demonstrate the first fit , best fit , worst fit and next fit allocation of given process queue and free list

Process queue

85 K	35 K	70 K	100 K

Free list

80K	130K	90 K	40 K

15. Apply Stack data structure in infix to postfix conversion

OR

16. Develop the algorithms for ENQUEUE(), and DEQUEUE() operations on queue data structures

17. Construct the algorithms for graph traversal(BFS, DFS)

OR

18. Explain 3 types of binary tree traversal

19. Explain Any 3 types of hash functions

OR

20. Represent the following values in the given order in a hash table (Size of hash table is 7 and hash function used is h(k)=k mod 7) for each of the scenario.

    19, 26, 13, 48, 17

    a) When collisions are handled by linear probing
    b) When collisions are handled by double using second hash function h=5-(k mod 5)

Syllabus

Module 1: Introduction to data structures (9 Hours)
Data Structures-Introduction and Overview- Arrays, Algorithm/Program Development, , Searching and Sorting.
Module 2: Linked lists (10 Hours)
Linked lists, singly linked list, Doubly linked list, Circular linked list, Applications of linked list, Dynamic Memory management.
Module 3 : Stacks and Queues (9 Hours)
Stack, Applications of stacks, Queues, Types of queues
Module 4 : Trees and graphs (10 Hours)
Trees, Binary Tree Traversals, Binary tree Applications, Graph, and Graph Applications.
Module 5 : Hash Table (7 Hours)
Hash Tables, Different Hash Functions, Collision Resolution Techniques, closed hashing and Open Hashing (Separate Chaining).

Text Books

T1. Samanta D., Classic Data Structures, Prentice Hall India, 2/e, 2009.

T2. Ellis horowitz, Sartaj Sahni, Fundamentals of Data structures, Galgotia Booksource

Reference Books

R1. Horwitz E., S. Sahni and S. Anderson, Fundamentals of Data Structures in C, University Press (India), 2008.

R2. Aho A. V., J. E. Hopcroft and J. D. Ullman, Data Structures and Algorithms, Pearson Publication,1983**.**

R3. Tremblay J. P. and P. G. Sorenson, Introduction to Data Structures with Applications, Tata McGraw Hill, 1995.

R4. Peter Brass, Advanced Data Structures, Cambridge University Press, 2008

R5. Lipschuts S., Theory and Problems of Data Structures, Schaum's Series, 1986.

R6. Wirth N., Algorithms + Data Structures = Programs, Prentice Hall, 2004.

R7. Hugges J. K. and J. I. Michtm, A Structured Approach to Programming, PHI, 1987.

R8. Martin Barrett, Clifford Wagner, And Unix: Tools For Software Design, John Wiley, 2008 reprint.

Course Contents and Lecture Schedule

	Module 1: Introduction to data structures	**9hrs**
1.1	**Data Structures-Introduction and Overview**: Definitions, Concept of data structure, classifications of data structure- ADT and CDT- Linear and nonlinear.	1
1.2	**Arrays:** definition, Representation of Single/Two dimensional arrays, Applications of array – searching –Sorting - Sparse Matrix- conversion of sparse matrix into 3 tuple form.	2
1.3	**Algorithm/Program Development:** Analysis of algorithms. Space Complexity, Time Complexity - Best case, worst case, average case. **Searching** : linear and binary search – Complexity Analysis (Detailed analysis is not required)	2
1.4	**Sorting**: classifications- Internal sorting – External sorting , N^2 Sorting : Selection, bubble and insertion- Complexity analysis (Detailed analysis is not required)	2
1.5	N log $_n$ Sorting : Quick Sort and Merge Sort (Recursive Algorithms)- Complexity Analysis (Detailed analysis is not required)	2
	Module 2: Linked lists	**10 hrs**
2.1	**Linked lists**: static and dynamic representation, Classification -Singly linked list- Doubly linked list- Circular linked list, array and linked list. **Singly linked list**: Operations on Singly linked list- Traversal-Insertion-deletion, copying -searching - Merging.	2
2.2	**Doubly linked list**: Operations on doubly linked list- Insertion-deletion.	2
2.3	**Circular Linked list** : Operations on circular linked list-Insertion and deletion	2
2.4	**Applications of linked list**: Polynomial representation and manipulation (addition)- Dynamic Memory management.	2
2.5	**Dynamic Memory management**: Fixed sized and variable sized memory allocation and de-allocation. First-fit, best-fit and worst-fit allocation schemes and problems.	2
	Module 3: Stacks and Queues	**9 hrs**
3.1	**Stack:** Definition, Schematic Diagram of stack, Array and Liked list representation of stack , operations on stack using array and linked list (PUSH(),POP(),STATUS()) .	2
3.2	**Applications of stacks**: Infix to postfix conversion- post fix evaluation, string reversal, delimiter matching.	3
3.3	**Queues**: Definition, Schematic Diagram of queue, Array and Liked list representation of queue , operations on queue using array and linked list (EQUEUE(),DEQUEUE(),STATUS()) .	2
3.4	**Types of queue** : circular queue-priority queue- doubly ended queue	2

	Module 4: Trees and graphs	**10 hrs**
4.1	**Trees:** Basic terminologies, Binary Trees, Properties of binary trees, linear and linked representations, Complete and full Binary Tree.	2
4.2	**Binary Tree Traversals**: Preorder -In order and post order (Recursive, non-recursive)-problems	1
4.3	**Binary tree Applications:** Expression tree creation, heap trees (concepts), Binary search tree – creation, insertion and deletion and search operations	3
4.4	**Graph**: Terminologies, set representations, linked/adjacency list representation, Adjacency matrix linear representation **Graph traversal**: Breadth First Search (BFS), Depth First Search (DFS) - related problems.	2
4.5	**Graph Applications:** Shortest Path Problem-Dijkstras Algorithm	2
	Module 5: Hash Table	**7 hrs**
5.1	**Hash Tables**-Hash Functions- Features of hash function.	1
5.2	**Different Hash Functions:** Division Method- Multiplication Method - Mid Square Method, Folding Method- related problems.	2
5.3	**Collision Resolution Techniques:** Closed hashing (Linear probing) and Open Hashing (Separate Chaining) . Closed hashing(Linear probing) -Drawbacks- Remedies - Radom Probing – Double hashing/Re-hashing –Quadratic Probing, problems to create hash tables using linear probing and Random probing, double hash and quadratic probing .	3
5.4	Open Hashing (Separate Chaining)	1

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT203	DIGITAL SYSTEM DESIGN	PCC	3	1	0	4

Preamble: The syllabus is prepared with the view of preparing the Engineering Graduates capable of understanding the basic digital logic design and implementation. All students of computing should acquire some understanding and appreciation of a computer system's functional components, their characteristics, their performance, and their interactions.

Prerequisite: NIL

Course Outcomes: After the completion of the course, the student will be able

CO No.	Course Outcome(CO)	Bloom's Category
CO 1	To perform base conversion and arithmetic operations in various number systems.	Apply
CO 2	To design digital circuits using simplified Boolean functions	Create
CO 3	To develop simple design of combinational circuits	Apply
CO 4	To develop simple design of sequential circuits	Apply
CO 5	To interpret the generalization of synchronous and asynchronous sequential circuits	Understand

Mapping of course outcomes with program outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3			3	-	-	-	-	-	-	-	2
CO 2	3	3	3	3	-	-	-	-	-	-	-	2
CO 3	3	3	3	3	-	2	-	1	-	-	-	2
CO 4	3			3	-	-	-	-	-	-	-	2
CO 5	3	3	3	3	2	2	-	1	-	-	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test 1 (Marks)	Test 2 (Marks)	
Remember	10	5	20
Understand	15	10	20
Apply	10	10	25
Analyse	10	10	15
Evaluate	5	10	10
Create		5	10

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	4 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1): To understand the basic concepts of Number systems

1. Convert the given number from decimal number system to binary, octal, and hexadecimal number system.
2. Perform Arithmetic operations on different number system.
3. Represent the different coding schemes.

Course Outcome 2 (CO2): To design digital circuits using simplified Boolean functions

1. Simplify the given expression using Postulates of Boolean algebra.

2. Convert a given expression to standard and canonical forms.
3. Simplify the given expression using Karnaugh Map or Quine –McClusky minimization technique.

Course Outcome 3 (CO3): To analyze and design combinational circuits

1. Analyse a given circuit and explain the results obtained by the circuit.
2. Design a Ccarry look ahead adder.
3. Design a four-bit magnitude comparator.

Course Outcome 4 (CO4): To understand the basics of sequential circuits

1. Understand the functioning of Latches and Flip Flops.
2. Design Master-Slave Flip Flops.
3. Understand the basics of different types of Flip Flops.

Course Outcome 5(CO5): To analyze and design synchronous and asynchronous sequential circuits

1. Analyse a given circuit and explain the results obtained by the circuit.
2. Implement a serial adder using a shift register.
3. Design and construct a 4-bit ring counter with only one flip-flop is clear at any particular time and all other flip-flops are set. Give its timing diagram.
4. Using an example, show the Race-Free State Assignment in an asynchronous sequential circuit.

Model Question Paper

Course Code: ITT203
Course Name: DIGITAL SYSTEM DESIGN

Max. Marks: 100 Duration: 3 Hours

PART A
Answer all questions, each carries 3 marks.

1. Convert $(76.75)_{10}$ to binary, octal and hexadecimal.
2. Determine the base of the numbers in the operation; $\mathbf{58/4}$= 15.
3. Simplify the Boolean expression to minimum number of literals.
   $F=\bar{B}C+A\bar{B}+AB\bar{C}+ABC\bar{D}+\bar{A}\bar{B}CD+A\bar{B}CD$
4. Find the complement of the Boolean function $F=\bar{A}+AB\bar{C}$.And prove that $F+\bar{F}=1$and $F.\bar{F}=0$.
5. Design a 4-to-2 line priority encoder.

6. Explain the difference between a latch and a flip-flop.

7. Give the characteristics equations for D, JK and T flip-flops.

8. Discuss in detail about Race condition.

9. With a neat diagram, discuss about SISO.

10. Design a 4-bit ring counter.

PART B

Answer all questions, each carries 14 marks.

11. a) Using Booth algorithm, perform multiplication of (-14) and (-7). (5)

b) Represent the unsigned decimal numbers 572.36 and 382.71 in BCD. Show the necessary steps to form their sum and difference. (9)

OR

12. a) (i) Find the decimal equivalent of $(A40F)_{16}$
(ii) Find the 16's complement of $(A40F)_{16}$
(iii) Convert to binary $(A40F)_{16}$
(iv) Finds the 2's complement of the result in (iii) (8)

b) Perform addition, subtraction, multiplication, and division of the following binary numbers without converting them to decimal : 1000110 and 110. (6)

13. a) For the Boolean function

$$F = w'xy' + xy'z + x'y'z + w'xy + wx'y + wxy$$

(i) Draw the logic diagram, using the original Boolean expression.
(ii) Simplify the Boolean algebra to a minimum number of literals.
(iii) Obtain the truth table of the function from the simplified expression and show that it is the same as the original Boolean expression. (9)

b) Prove that $A+\bar{A}B = A + B$ using Boolean postulates. (5)

OR

14. a) Simplify the following functions using Quine- McClusky method :

$$f(a,b,c,d) = \Sigma m\,(2, 3, 4, 5, 13, 15) + \Sigma d\,(8, 9, 10, 11).$$ (7)

b) Using K-map simplify following Boolean expression & give implementation of same using gates $F(A,B,C,D) = \Sigma\,(2,4,8,15) + \Sigma D(0,3,9,12)$ (7)

15. a) Design a combinational circuit to implement a 4-bit carry look-ahead adder. (7)

b) Design a 4-bit code-converter to convert BCD to gray code. (7)

OR

16. a) Implement the Boolean function f(w,x,y,z) = Σm(0,1,5,6,7,9,12,15) using 8-to-1 multiplexer. (7)
    b) Implement a 2-bit Magnitude comparator and write down its design procedure. (7)

17. a) For the following state table

Present State	Next State		Output	
	x=0	x=1	x=0	x=1
a	b	c	0	0
b	d	f	1	0
c	b	e	0	0
d	f	h	1	0
e	b	e	0	0
f	g	a	1	1
g	a	h	0	0
h	g	e	1	1

i. Draw the corresponding state diagram.
ii. Tabulate the reduced state table.
iii. Draw the state diagram corresponding to the reduced state table.
iv. Design the sequential circuit using flip-flops. [Hint:Unused states may be considered, as don’t cares.] (9)

b) Design D Flip Flop by using SR Flip Flop and draw the timing diagram. (5)

OR

18. a) Explain the state reduction in the sequential circuits using an example. (9)

    b) Draw the circuit of JK flip flop using NAND gates and explain its operation. (5)

19. a) Implement a four-bit universal shift register. Explain its design. (7)

    b) What do you mean by ripple counter? Design and implement a BCD ripple counter. (7)

OR

20. a) Tabulate the PLA programming table for the four Boolean functions listed below. Minimize the numbers of product terms.
    A(x, y, z) = Σm(1, 3, 5, 6)
    B(x, y, z) = Σm (0, 1, 6, 7)
    C(x, y, z) = Σm (3, 5)
    D(x, y, z) = Σm (1, 2, 4, 5, 7) (9)

    b) What are the operations that can be performed on a RAM? (5)

Syllabus

Module 1: NUMBER SYSTEM (9 Hours)
Number Systems – Decimal, Binary, Octal, Hexadecimal - conversion from one system to another – Representation of negative numbers using 2's compliment. Arithmetic Operations – Addition, Subtraction, Multiplication, Division of Binary numbers, Booths algorithm for multiplication, Representation of negative numbers, Representation of floating point numbers. Representation of BCD numbers, BCD Addition, Binary Codes – Gray codes – excess 3 code- Character Coding Schemes – ASCII, EBCDIC.
Module 2: BOOLEAN ALGEBRA & LOGIC GATES (9 Hours)
Boolean Algebra - Postulates of Boolean algebra - Canonical and Standard Forms - Simplification of Boolean Functions using Karnaugh Map - Product-of-Sums Simplification — Don't-Care Conditions – Quine –McClusky minimization technique – Basic Gates- Universal Gates.
Module 3: COMBINATIONAL LOGIC (9 Hours)
Combinational Circuits – Analysis and Design Procedures - Binary Adder-Sub tractor (Half & Full) - Carry look ahead adder, BCD adder, code converter, - Magnitude Comparator - Decoders – Encoders Parity Generator– Multiplexers – DE multiplexers – Implementation of Boolean functions using MUX.
Module 4: SEQUENTIAL LOGIC CIRCUITS (9 Hours)
Sequential Circuits - Storage Elements: Latches , Flip-Flops – RS, JK, D, T, Triggering of flip-flops, Master-Slave- Analysis of Clocked Sequential Circuits - Design Procedure-using JK,D & T.
Module 5: COUTERS AND SHIFT REGISTERS (9 Hours)
Registers - Shift Registers – SISO, PIPO, SIPO, PISO- Universal shift registers, Counters- Design of Counters- Synchronous & Asynchronous Counters — up-down counter, Decade counter, BCD counter, Johnson counter, Ring counter ,Memory & Programmable logic- RAM, ROM, PLA,PAL

Text Books

1. Mano M. M. and Michael D. Ciletti, *Digital Design*, 4/e, Pearson Education, 2013.
2. Thomas L. Floyd, *Digital Fundamentals*, 11th Edition, Pearson Education, 2015.
3. N. N. Biswas, *"Minimization of Boolean Functions,"* in *IEEE Transactions on Computers*, vol. C-20, no. 8, pp. 925-929, Aug. 1971. doi: 10.1109/T-C.1971.223373

Reference Books

1. Charles H Roth ,Jr, Lizy Kurian John, *Digital System Design using VHDL*,2/e, Cengage Learning
2. Mano M. M. and Michael D. Ciletti, Digital Design with an Introduction to the Verilog HDL, 5/e, Pearson Education, 2013.
3. Tokheim R. L., Digital Electronics Principles and Applications, 7/e, Tata McGraw Hill, 2007.
4. Rajaraman V. and T. Radhakrishnan, *An Introduction to Digital Computer Design*, 5/e, Prentice Hall India Private Limited, 2012.
5. Leach D, Malvino A P, Saha G, *Digital Principles and Applications,* 8/e, McGraw Hill Education, 2015.
6. M. Morris Mano, *Computer System Architecture*, 3/e, Pearson Education, 2007.

7. Harris D. M. and, S. L. Harris, *Digital Design and Computer Architecture*, 2/e, Morgan Kaufmann Publishers, 2013

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**NUMBER SYSTEM**	**9 Hours**
1.1	Number Systems – Decimal, Binary, Octal, Hexadecimal - conversion from one system to another – Representation of negative numbers using 2's compliment.	3 hours
1.2	Arithmetic Operations – Addition, Subtraction, Multiplication, Division of Binary numbers, Booths algorithm for multiplication, representation of negative numbers, Representation of floating point numbers.	4 Hours
1.3	Representation of BCD numbers , BCD Addition Binary Codes – Gray codes – excess 3 code- Character Coding Schemes – ASCII, EBCDIC	2 Hours
2	**BOOLEAN ALGEBRA & LOGIC GATES**	**9 Hours**
2.1	Boolean Algebra - Postulates of Boolean algebra - Canonical and Standard Forms	2 Hours
2.2	Simplification of Boolean Functions using Karnaugh Map - Product-of-Sums Simplification — Don't-Care Conditions	2 Hours
2.3	Quine –McClusky minimization technique	2 Hours
2.4	Basic Gates-Universal Gates.	3 Hours
3	**COMBINATIONAL LOGIC**	**9 Hours**
3.1	Combinational Circuits – Analysis and Design Procedures - Binary Adder-Subtractor - Carry look ahead adder, BCD adder	3 Hours
3.2	Code converter, - Magnitude Comparator - Decoders – Encoders – Multiplexers	3 Hours
3.3	Parity Generator– Multiplexers – DE multiplexers – Implementation of Boolean functions using MUX.	3 Hours
4	**SEQUENTIAL LOGIC CIRCUITS**	**9 Hours**
4.1	Sequential Circuits - Storage Elements: Latches , Flip-Flops – RS, JK, D, T, Triggering of flip-flops, race condition- Master-Slave	3 Hours
4.2	Analysis of Clocked Sequential Circuits	3 Hours
4.3	State Reduction and Assignment - Design Procedure- using JK,D & T	3 Hours
5	**COUTERS AND SHIFT REGISTERS**	**9 Hours**
5.1	Registers - Shift Registers – SISO, PIPO, SIPO, PISO- Universal shift registers	2 Hours
5.2	Design of Counters- Synchronous & Asynchronous Counters — up-down counter.	3 Hours
5.3	Counters-, Decade counter, BCD counter, Johnson counter, Ring counter	2 Hours
5.4	Memory & Programmable logic-RAM, ROM, PLA,PAL	2 Hour

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT205	**PROBLEM SOLVING USING PYTHON**	**PCC**	**3**	**1**	**0**	**4**

Preamble: The syllabus is prepared with the view of preparing the Engineering Graduates capable of writing readable PYTHON programs to solve computational problems that they may have to solve in their professional life. The course content is decided to cover the essential programming fundamentals which can be taught within the given slots in the curriculum. This course has got 3 lecture hours and 1 tutorial hour per week for learning and practicing programming using PYTHON. The instructor is supposed to give homework/assignments to write simple programs in the rough record as and when the required theory part is covered in the class. The programs that require time and effort can be done in the Lab sessions. The students are expected to come prepared with the required program written in the rough record for the lab classes.

Prerequisite: NIL

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcomes	Bloom's Category
CO 1	Write programs using Python and learn its execution environment	Understand
CO 2	Apply programs to implement various computational tasks which requires loops and conditional statements	Apply
CO 3	Write programs using functions and packages	Understand
CO 4	Apply programs to implement the concept of file handling using python	Apply
CO 5	Design object oriented programs to implement daily life problems and their solutions	Apply

Mapping of course outcomes with program outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	1	1	1	-	-	-	-	-	-	-	-	-
CO 2	2	2	1	1	-	-	-	-	-	-	-	-
CO 3	3	2	2	2	2	-	-	-	-	-	-	1
CO 4	1	1	1	2	1	-	-	-	-	-	-	1
CO 5	3	1	3	2	1	1	-	-	-	-	-	1

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	25
Understand	20	20	35
Apply	20	20	40
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):
Describe identifiers, variables, keywords, expressions and statements, Operators and operands.
Describe the expression evaluation in Python.
Describe the syntax of control statements in Python.
Write programs to solve problems using various control structures.
Differentiate between Break and Continue.

Course Outcome 2 (CO2):
Build applications of various string manipulations by using methods and functions available with string module in python.

Build applications of various List operations, including matrix representation.
Build applications of Tuple and various Tuple operations.
Build applications of Dictionary and its related operations, functions and methods.

Course Outcome 3(CO3):
Describe function definition and function access in python.
Differentiate between parameters and arguments.
Differentiate between type conversion and coercion.
Describe mathematical & lambda functions in Python.
Explain the concept of composition of functions in Python.
Explain Recursion and its implementation in Python.
Explain the concepts of modules and packages in Python. How and why import method is used.
Write programs to solve problems using the concept of functions and recursion.

Course Outcome 4 (CO4):
Apply the concept of file operations including opening, writing to and reading from files, and writing variables using Python.
Build applications to demonstrate the concept of Pickling.
Build programs to demonstrate the concept of Exception Handling in python.

Course Outcome 5 (CO5):
Build classes using python & Access class variables.
Build programs in Python to demonstrate the use of instances as arguments and return values.
Build programs in Python to demonstrate the concept of Constructors, class attributes and destructors.
Model the concept of Inheritance using Python.

Model Question paper

Course Code: ITT 205

Course Name: Problem Solving Using Python

Max.Marks:100 **Duration: 3 Hours**

PART A

Answer all Questions. Each question carries 3 Marks

1. Write a python program to find the sum of all odd terms in a group of n numbers entered by the user.
2. What is the use of *pass* statement in Python?
3. Write a Python code to check whether two strings are equal or not.
4. Write a Python code to search an element in a list.
5. List the advantages of using functions in a program.
6. State the use of dump method with suitable example.

7. Write a function exists() which returns True if the given file exists and False if it does not.
8. Why exceptional handling is required in programming?
9. Describe the concept of Constructor with an example.
10. Explain the purpose of __init__() method in Python.

(10x3=30)

Part B

Answer any one Question from each module. Each question carries 14 Marks

11. **(a)** Write a Python program to find the largest and second largest of n numbers. Assume n>=3 and all the numbers are distinct. No sorting algorithm should be used. (7)

(b) What are arithmetic operators used in Python? Explain each using an example (7)

OR

12. **(a)** Write a Python program to print the odd composite numbers between m and n, where m and n are positive integers greater than 1. (8)

(b) Define the following
i) Atoms
ii) Identifiers
iii) Literals (6)

13. **(a)** Write a Python code to add two matrices using list. (6)

(b) Write a Python program to reverse a string and print whether its palindrome or not. (8)

OR

14. **(a)** How to create Dictionary in Python? Write a Python program to read and display a sparse matrix using dictionary. (7)

(b) Write a program to
i) convert all small letters in a string into capital letters
ii) find the occurrence of a given substring. (7)

15.(a) Compare the built-in functions int() and str() with examples. (4)

(b) Write a program using function to display a multiplication table of n*n size, for any given 'n'. (5)

(c) Write a program using function to display a find the binomial coefficient, ${}^{n}c_{r}$. (5)

OR

16 (a). What is recursion? Write a recursive function to find the factorial of a number. (6)

(b) Write a program using function to check the type of a triangle by getting the values from the user. (8)

17. **(a)** How exceptions are handled in Python? Illustrate with example. (10)

(b) Write a program to read numbers sorted in one file and store the sorted numbers in another file after deleting duplicates. (4)

OR

18. (a). Describe the use of try-except method in Python with suitable Illustrations. (6)

(b) Write a Python code to read a text file, copy the contents to another file after removing the blank lines. (8)

19. (a). Write a Python code to create a class named 'Member' having the following members: Data members Name, Age, Phone number, Address, Salary. It also has a method named 'printSalary' which prints the salary of the members. Two classes 'Employee' and 'Manager' inherits the 'Member' class. The 'Employee' and 'Manager' classes have data members 'specialization' and 'department' respectively. Now, assign name, age, phone number, address and salary to an employee and a manager by making an object of both of these classes and print the same. (8)

(b) Create a class person with attributes Name, age, salary and a method display() for showing the details. Create two instances of the class and call the method for each instance. (6)

OR

20. (a) Define the terms class, attribute, method and instance with the help of an example. (4)

(b) Write a Python program to find out the total mark of a student using multiple inheritance. Declare a Student class. Student class should have the member functions for accept and display student details. Declare a Sports class to accept and display sports marks of the student. Derive a class statement from both the classes. This class should have the member functions to accept marks of three subjects and add those marks with sports marks and display the result. (10)

(14 x 5=70)

Syllabus

Module 1	**9 hours**
Introduction To Python: Understanding Python-identifiers, variables, keywords, expressions and statements, evaluation of expressions, Operators and operands, operator precedence, indentation. **Python Program Flow Control:** Decision making- if, if..else, elif. Loops - for, while, for...else, while...else, Control statements using pass, continue, break.	
Module 2	**9 hours**
Strings and lists – string traversal, string slices and comparison with examples, The string module, character classification. List- List values, accessing elements, list membership, Lists and for loops, List operations, List slices, List deletion, Matrices. Tuples - mutability and tuples, tuple assignment, Tuples as return values, Tuple operations. Dictionaries – operations and methods.	
Module 3	**9 hours**
Python Functions, Modules and Packages: Function definition, calling functions, parameters and arguments, the return statement, type conversion and coercion, composition of functions, Lambda function, mathematical functions, user-defined functions, Recursion, Modules- Built-in modules, creating modules, import statement. Packages in Python - importing modules from a package.	

Module 4	9 hours
Python Files and exceptions: Python file handling, open, write, read text files, writing variables, Directories in Python, Pickling, Exception Handling.	
Module 5	**9 hours**
Python Object Oriented Programming: Introduction to classes and objects - class definition, attributes, instances, sameness, instances as arguments and return values. Constructor, class attributes and destructors, Inheritance.	

Text Books

1. Allen Downey, Jeffrey Elkner, Chris Meyers, “ How to think like a Computer Scientist-Learning with Python”, Green Tea Press, First edition, 2002.
2. Mark Lutz,”Learning Python: Powerful Object-Oriented Programming” , O’Reilly Media Inc.,5th,2013

Reference Books

1. Kenneth A. Lambert, B. L. Juneja, “Fundamentals of Python”, Cengage Lerning India Pvt. Ltd., 2015.
2. S.A.Kulkarni, “Problem Solving and PYTHON Programming”, 2nd edition, Yes Dee Publishing Pvt Ltd, 2018
3. Mark Summerfield,”Programming in Python 3: A Complete Introduction to the Python Language”, Pearson Education, 2nd,2018
4. Yashavant Kanetkar ,Aditya Kanetkar ,”Let Us Python ”,BPB Publications, 1st Edition, 2019
5. Allen Downey, “Learning with Python”, Dreamtec Press, 1st Edition, 2015
6. https://docs.python.org/3/reference/

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction To Python:**	**9 hours**
1.1	Understanding Python-identifiers, variables, keywords, expressions and statements.	2
1.2	Evaluation of expressions, Operators and operands, operator precedence, indentation	1

1.3	Python Program Flow Control: Decision making- if, if..else, elif.	2
1.4	Loops - for, while, for...else, while...else	2
1.5	Control statements using pass, continue, break.	2
2	**Strings and lists:**	**9 hours**
2.1	String traversal, string slices and comparison with examples	1
2.2	The string module, character classification.	1
2.3	List- List values, accessing elements, list membership, Lists and for loops, List operations, List slices, List deletion	2
2.4	Matrices	1
2.5	Tuples- mutability and tuples, tuple assignment, tuples as return values, Tuple operations.	2
2.6	Dictionaries – operations and methods.	2
3	**Python Functions, Modules And Packages:**	**9 hours**
3.1	Function definition, calling functions, parameters and arguments, the return statement.	1
3.2	Type conversion and coercion, composition of functions	1
3.3	Lambda function, mathematical functions	1
3.4	user-defined functions	1
3.5	Recursion	1
3.6	Modules -Built-in modules	1
3.7	Creating modules, import statement.	1
3.8	Packages in Python - importing modules from a package.	2
4	**Python Files and exceptions:**	**9 hours**
4.1	Python file handling, open, write, read text files	4
4.2	Writing variables	1
4.3	Directories in Python	1
4.4	Pickling	1
4.5	Exception Handling.	2

5	**Python Object Oriented Programming:**	**9 hours**
5.1	Introduce classes and objects	1
5.2	Class definition, attributes, instances, sameness	1
5.3	Instances as arguments and return values.	1
5.4	Constructor	2
5.5	Class attributes and destructors	2
5.6	Inheritance	2

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITL201	DATA STRUCTURES LAB	PCC	0	0	3	2

Preamble:

This lab is intended to make the students capable of

- understanding the importance of data structures, abstract data type, and their basic usability in different application,
- Implementing linear and non-linear data structures using linked lists and arrays.
- Applying various data structure such as stacks, queues, trees, graphs, etc. to solve various computing problems
- Identifying suitable data structure and algorithm to solve a real world problem.

Prerequisite: programming in C

Course Outcomes: After the completion of the course the student will be able to

CO_No	Course Outcome(CO)	Bloom's Category
CO 1	**Compare** various kinds of searching and sorting techniques	level 5: Evaluate
CO 2	**Construct** Linear and nonlinear data structures using arrays and linked list	level 6:Create
CO 3	**Develop** Programs employing dynamic memory management	level 6:Create
CO 4	**Choose** appropriate data structure to solve various computing problems.	level 5: Evaluate
CO 5	**Originate** hash tables and collision resolution Techniques	level 6:Create
CO 6	**Identify** suitable data structure and algorithm to solve a real world problem.	level 3:Apply

Mapping of course outcomes with program outcomes

COs	PROGRAMME OUTCOMES (PO)											
	PO 1	4 PO 2	5 PO 3	5 PO 4	6 PO 5	3 PO 6	2 PO 7	3 PO 8	3 PO 9	3 PO 10	3 PO 11	3 PO 12
CO 1	3	3	3	2	2	1	-	-	-	-	2	1
CO 2	3	3	3	3	3	1	-	-	-	-	2	1
CO 3	3	3	3	3	3	1	-	-	-	-	2	1
CO 4	3	3	3	3	3	1	-	-	-	-	2	1
CO 5	3	3	3	3	3	1	-	-	-	-	2	1
CO 6	3	3	3	3	3	1	-	-	-	-	2	1

3/2/1: high/medium/low

Assessment Pattern

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	75	75	2.5 hours

Continuous Internal Evaluation Pattern:

Attendance : 15 marks
Continuous Assessment : 30 marks
Internal Test (Immediately before the second series test): 30 marks

End Semester Examination Pattern: The following guidelines should be followed regarding award of marks
(a) Preliminary work : 15 Marks
(b) Implementing the work/Conducting the experiment : 10 Marks
(c) Performance, result and inference (usage of equipments and trouble shooting) : 25 Marks
(d) Viva voce : 20 marks
(e) Record : 5 Marks

General instructions: Practical examination to be conducted immediately after the second series test covering entire syllabus given below. Evaluation is a serious process that is to be conducted under the equal responsibility of both the internal and external examiners. The number of candidates evaluated per day should not exceed 20. Students shall be allowed for the University examination only on submitting the duly certified record. The external examiner shall endorse the record.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Develop a C program to implement insertion sort, selection sort and bubble sort.
2. Design a program to Implement i) Quick sort ii) Merge sort.
3. Create programs for i) Linear Search ii) Binary Search.

Course Outcome 2 (CO2)

1. Design a Menu driven program to implement singly linked list operations with options for insertion, deletion, search and traversal.
2. Develop Menu driven program to implement doubly linked list operations with options for insertion at front, insertion at end, deletion at front, deletion at end and traversal.

Course Outcome 3(CO3):

1. Design a Menu driven program to implement singly linked list operations with options for insertion, deletion, search and traversal.
2. Simulate first fit, best fit and worst fit memory allocation strategies using linked list.

Course Outcome 4 (CO4):

1. Device Dijkstra's Algorithm for finding Shortest path
2. Apply Queue and stack in Breadth First Search and Depth First Search respectively

Course Outcome 5 (CO5):

1. Implement hash table using various mapping functions
2. Resolve the collisions if any using collision resolution techniques like linear Probing, Random Probing, Double hashing and Quadratic Probing

Course Outcome 6 (CO6):

1. Design and implement an application program which is to be used in Ticket counter, where First person gets ticket first and go out first, using suitable data structure. The program should do the following functions

   a) When new person come to the counter the details of the person (Name and age) should be added to the data structure.

   b) After issuing tickets, the details of the corresponding person should be deleted from the data structure

   (Hint: Each node of linked list should contain fields Name, Age and Rink)
2. Design and implement an application program to store the details of pass percentage of the college in chronological order of years (oldest to newest) and retrieve the information in reverse chronological order of years, using suitable data structure.

   Hint: A Node of linked list may contain the fields Year, Pass Percentage and Rlinlk.

   Menu may contain

   a) Store details

   b) Retrieve details

LIST OF EXPERIMENTS

1. Develop a C program to implement insertion sort, Selection sort and bubble sort*.
2. Design a program to Implement i) Quick sort ii) Merge sort*.
3. Create Programs for i) Linear Search ii) Binary Search*.
4. Create a menu driven program to implement singly linked list operations with options for insertion, deletion, search and traversal*.
5. Device a menu driven program to implement doubly linked list operations with options for insertion at front, insertion at end, deletion at front, deletion at end and traversal.
6. Apply linked list concept to perform polynomial addition*
7. Simulate first fit, best fit and worst fit memory allocation strategies using linked list*.
8. Develop a program to perform stack operations using i) array ii) linked list*.
9. Perform queue operations using i) array ii) linked list*.

10. Apply stack to perform i) Infix to postfix conversion ii) Postfix evaluation *
11. Develop a program to perform preorder, in-order, post order traversals on binary trees*
12. Construct binary search trees to perform insertion, deletion, search
13. Apply Queue and stack in Breadth First Search and Depth First Search respectively *
14. Device Dijkstra's Algorithm for finding Shortest path
15. Resolve the collisions if any using collision resolution techniques like linear Probing, Random Probing, Double hashing and Quadratic Probing*
    (* indicates mandatory experiments.)

Text Books

1. Samanta D., Classic Data Structures, Prentice Hall India, 2/e, 2009.
2. Ellis horowitz, Sartaj Sahni, Fundamentals of Data structures, Galgotia Booksource

Reference Books

1. Horwitz E., S. Sahni and S. Anderson, F
2. undamentals of Data Structures in C, University Press (India), 2008.
3. Aho A. V., J. E. Hopcroft and J. D. Ullman, Data Structures and Algorithms, Pearson Publication,1983.
4. Tremblay J. P. and P. G. Sorenson, Introduction to Data Structures with Applications, Tata McGraw Hill, 1995.
5. Peter Brass, Advanced Data Structures, Cambridge University Press, 2008
6. Lipschuts S., Theory and Problems of Data Structures, Schaum's Series, 1986.
7. Wirth N., Algorithms + Data Structures = Programs, Prentice Hall, 2004.
8. Hugges J. K. and J. I. Michtm, A Structured Approach to Programming, PHI, 1987.
9. Martin Barrett, Clifford Wagner, And Unix: Tools For Software Design, John Wiley, 2008 reprint.

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITL203	PROGRAMMING AND SYSTEM UTILITIES LAB	PCC	0	0	3	2

Preamble: This laboratory course is meant for understanding the fundamental system utilities. The course is also aimed for understanding and practicing the programming language Python.

Prerequisite: PYTHON programming knowledge and Computer Fundamentals.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcomes	Bloom's Category
CO 1	Develop readable* Python programs by making use of basic constructs- Decision controls, Looping controls, Lists, Tuple and Strings	Create
CO 2	Design modular Python programs using normal and recursive functions	Create
CO3	Design programs using Dictionaries and Files	Create
CO 4	Experiment with the basic Windows/ Linux administration & network configuration utilities	Apply
CO 5	Experiment with version control tools using git	Apply
readable* - readability of a program means the following: 1. Logic used is easy to follow 2. Standards to be followed for indentation and formatting 3. Meaningful names are given to variables 4. Concise comments are provided wherever needed		

Mapping of course outcomes with program outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	2	3	1	3	-	-	-	-	-	-	-
CO 2	2	3	3	3	3	-	-	-	-	-	-	-
CO 3	2	3	3	3	3	-	-	-	-	-	-	-
CO 4	2	2	1	2	1	-	-	-	-	-	-	-
CO 5	-	-	2	-	3	-	-	-	3	2	-	-

3/2/1: High/Medium/Low

Assessment Pattern

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	75	75	2.5 hours

Continuous Internal Evaluation Pattern:

Attendance : 15 marks
Continuous Assessment : 30 marks
Internal Test (Immediately before the second series test) : 30 marks

End Semester Examination Pattern: The following guidelines should be followed regarding award of marks

(a) Preliminary work : 15 Marks
(b) Implementing the work/Conducting the experiment : 10 Marks
(c) Performance, result and inference
(Usage of equipments and troubleshooting) : 25 Marks
(d) Viva voce : 20 marks
(e) Record : 5 Marks

General instructions: Practical examination to be conducted immediately after the second series test covering entire syllabus given below. Evaluation is a serious process that is to be conducted under the equal responsibility of both the internal and external examiners. The number of candidates evaluated per day should not exceed 20. Students shall be allowed for the University examination only on submitting the duly certified record. The external examiner shall endorse the record.

Course Level Assessment Questions

Course Outcome 1 (CO1):

Develop a python program to

1. Print all prime numbers with in an interval
2. Search an element in a list
3. Input a list of n numbers. Calculate and display the average of numbers. Also display the square of each value in the list
4. Add two matrices.
5. Find the number of occurrences of a given substring in a string.
6. Count the number of vowels, consonants, words and question marks in a given string.

Course Outcome 2 (CO2):

Develop a python program to

1. Find the value of **`nCr`** using function.
2. Implements calculator with functions like add, subtract, multiply, divide, exponent etc.
3. Find factorial of a given number using recursion.
4. Find n^{th} Fibonacci number using recursion.

Course Outcome 3 (CO3):

1. Develop a python program to create dictionary of phone numbers and names of n persons. Display the contents of the dictionary in alphabetical order of their names

2. Develop a Python program to implement the following scenario. A book shop details contains the Title of book and Number of copies of each title. As books are added to shop the number of copies in each should increase and as books are sold the number of copies in each should decrease.
3. Develop a python code to read a text file, copy the contents to another file after removing the blank lines.
4. Develop a python program to implement the following scenario. Given a file "`data.txt`" with three columns of data separated by spaces. Read it into 3 separate simple sequences.
5. Create a class student with attributes name, rollno and a method showData() for showing the details. Create two instances of the class and call the method for each instance. Develop a python program to implement the scenario.

Course Outcome 4 (CO4):
Perform the following operations:
1. Apply the use of ATTRIB windows command to change the attributes of a file.
2. Create a file **`xyz.txt`** and change the ownership of this file to some other user on your machine.
3. Create a file **`hello.txt`** and make it executable.
4. Create a new user account and home directory called "Duck" and Set the user account "Duck's" expiry date as 07 - 07 - 2020
5. Check the network connectivity of your computer using suitable Linux commands

Course Outcome 5 (CO5):
Perform the following:
1. Create a directory in your machine and make it as a repository and perform the following
   a. Create a text file and add some content into it.
   b. Add the file to the staging area of the Git repository.
   c. Commit the file to your repository.
   d. See the commit details using git log command.
2. Go to your Git repository and perform the following
   a. Do some modifications in your text file. Commit the changes.
   b. Try to revert to your old revision, again do some modifications in your text file and try to discard the changes.

<u>List of Experiments</u>

<u>Part A : Programming in Python</u>

1. **Basic programming experiments** to familiarization of data types and input-output statements
2. **Decision making, branching and looping statements**
3. **Function & Function calls**
   1. Function definitions and access
   2. Parameters and arguments
   3. Recursion
4. **Strings**
   a) String traversal, join, slicing
   b) String searching, Comparison

c) Other important String methods

5. **Lists, Tuples and Dictionaries**
    a) Creation of List & List Operations
    b) Tuple and Tuple operations
    c) Creation of Dictionary and Operations
    d) Comparison of List and Tuple
6. **Matrix representation**
    a) Creating matrix
    b) Matrix operations - addition, subtraction and multiplication
7. **Files and Operations**
    a) Files - defining, opening/closing, read/write operations
    b) Exceptions in Python
    c) Pickling
8. **Object Oriented Programming using Python**
    c) Creation of Classes & Instances, method calling
    d) Constructor & Destructor concepts
    e) Implementation of Inheritance

Part B : System Utilities

Basic Windows/Linux Administration Utilities

1. **Experiments on Windows Operating System**

    a. Perform the following commands

    DIR, TYPE, DEL, ERASE, MD, CD, COPYCON, RMDIR, REN, VER, DATE, TIME, TREE, PATH, CLS, RMDIR, BREAK, SET, EXIT, APPEND, CHKDISK, ATTRIB, SYS, EDIT, XCOPY, DISKCOPY

    b. Explore and describe some system utility like **regedit**, memory partioning, control panel and window tools

2. **Experiments on Linux Operating System**

    a) Perform general purpose utilities in Linux:

    echo, uname, whoami, passwd, date, date +%T, date +%h, date +%m, date +%y, date +"%h%y", cal, cal 12 2030, echo $HOME, pwd, ls, ls –all, ls –l, cat, cat > file1, cat >> file2, ls -l >fileinfo

    b) Familiarize working with files and managing file attributes

3. **Network Configuration Utilities**
    a) **ifconfig** utility, enable/disable network interface, **traceroute**,

    telnet, **nslookup, netstat, w, scp**, etc

    b) Connecting to the internet

4. **GIT for version control**
   a. Installation and configuration of Git on Ubuntu and Windows operating systems
   b. Perform Basic Git Commands (**`gitinit, add, status, commit`**, and **`log`**) and Git **`checkout`** command

Text Books

1. Allen Downey, Jeffrey Elkner, Chris Meyers, " How to think like a Computer Scientist-Learning with Python", Green Tea Press, First edition, 2002.
2. Mark Lutz,"Learning Python: Powerful Object-Oriented Programming" , O'Reilly Media Inc.,5th,2013

Reference

1. S.A.Kulkarni, "Problem Solving and PYTHON Programming", 2nd edition, Yes Dee Publishing Pvt Ltd, 2018
2. Kenneth A. Lambert, B. L. Juneja, "Fundamentals of Python", Cengage Learning India Pvt. Ltd., 2015.
3. Mark Summerfield," Programming in Python 3: A Complete Introduction to the Python Language", Pearson Education, 2nd,2018
5. Yashavant Kanetkar ,Aditya Kanetkar ,"Let Us Python ",BPB Publications, 1st Edition, 2019
6. Allen Downey, "Learning with Python", Dreamtec Press, 1st Edition, 2015
7. https://docs.python.org/3/reference/
8. Version Control with Git: Powerful tools and techniques for collaborative software development 2nd Edition, Kindle Edition by Jon Loeliger, Matthew McCullough
9. https://spoken-tutorial.org/

SEMESTER -3

MINOR

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT281	JAVA PROGRAMMING	VAC	3	1	0	4

Preamble: The syllabus is prepared with the intended to deliver students the elementary concepts of Java Programming and equip them to code java application built over those concepts. It also introduces to them advanced level areas like event driven programming with Java.

Prerequisite: Basics of Programming

Course Outcome (CO): After completion of the course, the student will be able to

CO No.	Course Outcome	Bloom's Category
CO1	Summarize Object Oriented Programming concepts and basic characteristics of Java	Understand
CO2	Summarize basic java packages, inheritance and interfaces	Understand
CO3	Summarize exceptions and I/O streams concepts	Understand
CO4	Demonstrate the usage of threads and generics classes	Understand
CO5	Build simple Graphical User Interface programs with Java	Apply

Mapping of Course Outcomes with Program Outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	2	2	-	1	-	-	-	-	-	-	-
CO2	1	1	2	1	-	-	-	-	-	-	-	-
CO3	3	1	1	-	-	-	-	-	-	-	-	-
CO4	3	1	1	1	-	-	-	-	-	-	-	-
CO5	3	1	1	2	2	-	-	-	-	-	-	-

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests (Marks)		End Semester Examination (Marks)
	1	2	
Remember	15	15	30
Understand	25	25	50
Apply	10	10	20
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Test (2 numbers) : 25 marks

Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain core principles of Object Oriented Design.
2. Describe the characteristics of Java.
3. Illustrate the concept of Polymorphism with an example.
4. Describe Java Source File Structure.
5. Explain constructors in detail

Course Outcome 2 (CO2):

1. Explain the concept of Super classes and sub classes.
2. Differentiate classes and interfaces.
3. Illustrate the concept of Interfaces with an example.
4. Describe Final methods and classes in java.
5. Explain extending interfaces in Java.

Course Outcome 3 (CO3):

1. Explain throwing and catching exceptions in Java.
2. Describe the different built-in exceptions in Java.
3. Illustrate the concept of Exception handling with an example.
4. Describe Byte and character streams in java.
5. Explain stack trace elements.

Course Outcome 4 (CO4):

1. Explain the concept of multithreading in Java.
2. Describe the different bounded types in Java.
3. Illustrate thread synchronization with example.
4. Describe inter-thread communication in java.
5. Explain the concept of generic programming in Java.

Course Outcome 5 (CO5):

1. Experiment with AWT in Java.
2. Build AWT programs for simulating Calculator.
3. Experiment with different Form components in Swing.

Model Question Paper

PART A

Answer all questions, each carries 3 marks

1. Why are java programs said to be platform independent?
2. Explain how access modifiers are used to control the visibility of identifiers?
3. What is the use of interface in java? Give example.
4. What is a package? How a class within a package is compiled and executed?
5. List out different exception classes in Java.
6. Write a note on byte stream and character stream classes in java
7. List out different bounded types in java
8. Draw the life cycle of a thread showing different stages and methods invoked.
9. List any 3 event sources and their corresponding generated event type and listeners used
10. What are the advantages of using swing?

PART B

11.
   a. What is the use of constructor in java? Give examples for different types of constructors. (7)
   b. Define a java class having overloaded methods to calculate the area of a rectangle and circle. (7)

OR

12.
   a. Write a Java program that counts the number of odd and even numbers in an array of 10 integers. (8)
   b. Briefly explain the architecture of JVM. (6)

13.
   a. List any five methods of String class, give examples. (5)
   b. With the help of examples, explain how inheritance is implemented in java. (9)

OR

14.
   a. Write a java program to count the number of occurrences of a particular word in a sentence using string handling methods. (8)
   b. Differentiate between abstract class and interface. (6)

15.
   a. Explain exception handling in java. Briefly explain various exception handling keywords in java with examples(9)
   b. Write a Java program that counts the number of words in a text file. (6)

OR

16.
   a. Write a Java program that accepts N integers through console and sort them in ascending order.(8)
   b. Explain the scenario under which the following three exceptions occur, NumberFormatException, ArrayIndexOutOfBoundsException and ArithmeticException (6)

17.
   a. Explain the different ways of creating threads in java (9)
   b. Explain the concept of generic programming in Java. (5)

OR

18.
   a. Write a java program to create two threads, one for writing even numbers and the other for writing odd numbers upto 100 in two different files. (12)
   b. What are the uses of synchronized keyword?(2)

19.
   a. Demonstrate the usage of any four Form components in Swing with an example. (4)
   b. Implement a Java AWT program for simulating Calculator(10)

OR

20. Write a java AWT based java program to display Fibonacci numbers in a list control upto a limit entered using TextField.The event handling as well as code for clearing the

components must be provided (14)

Syllabus

MODULE 1: INTRODUCTION TO OOP AND JAVA FUNDAMENTALS (10 Hours)
Object Oriented Programming – Abstraction – objects and classes – Encapsulation- Inheritance – Polymorphism- OOP in Java – Characteristics of Java – The Java Environment – Java Virtual Machine- Java Source File Structure – Compilation. data types, operators, control statements, Introduction to Java programming.– Classes fundamentals, objects, methods, constructors, parameter passing, overloading, access control keywords– static members -Comments, Arrays-Java Documentation usage
MODULE 2: INHERITANCE AND INTERFACES (10 Hours)
Inheritance – Super classes- sub classes –Protected members – constructors in sub classes- the Object class – abstract classes and methods- final methods and classes – Interfaces – defining an interface, implementing interface, differences between classes and interfaces and extending interfaces – Defining and importing packages. Strings-Java Built-in Classes and it's usage
MODULE 3: EXCEPTION HANDLING AND I/O (7 Hours)
Exceptions – exception hierarchy – throwing and catching exceptions – built-in exceptions, creating own exceptions, Stack Trace Elements. Input / Output Basics – Streams – Byte streams and Character streams – Reading and Writing Console – Reading and Writing Files
MODULE 4: MULTITHREADING AND GENERIC PROGRAMMING (8 Hours)
Differences between multithreading and multitasking, thread life cycle, creating threads, synchronizing threads, Inter-thread communication. Generic Programming – Generic classes – generic methods – Bounded Types – Restrictions and Limitations
MODULE 5: EVENT DRIVEN PROGRAMMING (10 Hours)
Graphics programming – Frame – Components – working with 2D shapes – Using color, fonts, and images – Basics of event handling – event handlers – adapter classes – actions – mouse events – AWT event hierarchy – Introduction to Swing – layout management – Swing Components – Text Fields , Text Areas – Buttons- Check Boxes – Radio Buttons – Lists- choices- Scrollbars – Windows –Menus – Dialog Boxes.

Text Books

1. Herbert Schildt, ―Java The complete reference‖, 8th Edition, McGraw Hill Education, 2011.
2. Cay S. Horstmann, Gary cornell, ―Core Java Volume –I Fundamentals‖, 9th Edition, Prentice Hall, 2013.

Reference Books

1. Paul Deitel, Harvey Deitel, ―Java SE 8 for programmers‖, 3rd Edition, Pearson, 2015.
2. Steven Holzner, ―Java 2 Black book‖, Dreamtech press, 2011.
3. Timothy Budd, ―Understanding Object-oriented programming with Java‖, Updated Edition, Pearson Education, 2000.

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**Introduction**	**10**
1.1	**Fundamentals of Object Oriented Programming:** Abstraction, objects and classes, Encapsulation, Inheritance, Polymorphism	3
1.2	**OOP in Java**: Characteristics of Java,The Java Environment – Java Virtual Machine-, Java Source File Structure , Compilation, data types, operators, control statements	3
1.3	**Introduction to Java programming:** Classes fundamentals, objects, methods, constructors, parameter passing, overloading, access control keywords, static members ,Comments, Arrays, -Java Documentation usage	4
2	**Inheritance And Interfaces**	**10**
2.1	**Inheritance:** Super classes, sub classes, Protected members, – constructors in sub classes, the Object class, abstract classes and methods, final methods and classes	5
2.2	**Interfaces**: defining an interface, implementing interface, differences between classes and interfaces and extending interfaces, Defining and importing packages. Strings, Java Built-in Classes and it's usage	5
3	**Exception Handling And I/O**	**7**
3.1	**Exceptions:** exception hierarchy, throwing and catching exceptions, built-in exceptions, creating own exceptions, Stack Trace Elements.	4
3.2	**Input / Output Basics:** Streams, Byte streams and Character streams, Reading and Writing Console, Reading and Writing Files	3
4	**Multithreading And Generic Programming**	**8**
4.1	**Multithreading:** Differences between multi threading and multitasking, thread life cycle, creating threads, synchronizing threads, Inter-thread communication.	4

4.2	**Generic Programming:** Generic classes, generic methods, Bounded Types, Restrictions and Limitations	4
5	**Event Driven Programming**	**10**
5.1	**Graphics programming:** Frame, Components, working with 2D shapes, Using color, fonts, and images	3
5.2	**Basics of event handling:** Event handlers, adapter classes, actions, mouse events, AWT event hierarchy	3
5.3	**Introduction to Swing:** layout management, Swing Components, Text Fields , Text Areas, Buttons, Check Boxes, Radio Buttons, Lists, choices, Scrollbars, Windows, Menus, Dialog Boxes.	4

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT283	DATA COMMUNICATION	VAC	3	1	0	4

Preamble: The syllabus is prepared in view of devising students capable of understanding the essential concepts and terminology used for data communication.

Prerequisite: NIL

Course objectives

- To introduce basic terminology and concepts used in data transmission
- To understand encoding techniques used in data communication
- Familiarize students the fundamental knowledge about computer networks

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome(CO)	Bloom's Category
CO 1	**Describe** the fundamental concepts of data communication, network models, and standards and wired networks.	Understand
CO 2	**Identify** different transmission media, data and signals.	Apply
CO 3	**Demonstrate** different encoding techniques used for analog to digital conversion and vice versa.	Understand
CO 4	**Describe** the different types of bandwidth utilization techniques and basic principles of switching.	Understand
CO 5	**Describe** the different access methods, channelization and wireless networks.	Understand

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	2	-	-	-	-	-	-	-	-	-	1
CO 2	1	2	-	2	2	-	-	-	-	-	-	1
CO 3	2	1	1	-	-	-	-	-	-	-	-	1
CO 4	3	2	2	1	1	-	-	-	-	-	-	1
CO 5	1	-	1	-	2	-	-	-	-	-	1	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	30
Understand	20	20	50
Apply	20	20	20
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; **Part A** and **Part B**. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer *all* questions. Part B contains 2 questions from each module of which student should answer *any one*. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain the fundamentals of data communication.

2. Explain the different types of networks.

3. Explain the different types of network models and standards.

4. Explain the different types of wired networks.

Course Outcome 2 (CO2)

1. Identify the different types of transmission media.

2. Identify the different types of data and signals.

3. Identify data rate limits for noisy and noiseless channel.

Course Outcome 3 (CO3):

1. Explain the fundamentals of data transmission.

2. Describe about different transmission modes.

3. Demonstrate the analog data conversion to analog signals and analog data into digital signals.

4. State and explain different keying techniques.

Course Outcome 4 (CO4):

1. Describe the functionality of different multiplexing techniques.

2. Illustrate about different spread spectrum techniques.

3. Describe basic principles of switching.

Course Outcome 5 (CO5):

1. Illustrate the random access methods used for collision avoidance.

2. Explain different controlled access methods.

3. Describe about different channelization protocols.

4. Demonstrate different services provided by wireless networks

Model Question Paper

PART A **(10*3=30)**

(Each Question carries 3 Marks)

1. List out the key components of data communication.

2. Summarize the essential criteria's required for an ideal network?

3. Mention Shannon's Theorem. Find out the channel capacity of a noisy channel which is having signal to noise ratio almost zero.

4. Analyze different characteristics used to measure the network performance.

5. Define the role of scrambling in digital transmission.

6. Calculate the baud rate for the given bit rate and type of modulation

    a. 4000 bps,QPSK

b. 36000 bps, 64-QAM

7. Identify the different phases used for communication in circuit switched network.
8. Define frequency division multiplexing and mention its applications.
9. Compare and contrast Pure ALOHA and Slotted ALOHA.
10. Examine the significance of transmission convergence sublayer in WiMax.

PART B **(5*14=70)**

11. Illustrate different types of connections and topologies used in network for connecting devices with the help of diagram.

OR

12. Substantiate the need of OSI model in network communication? Briefly explain the functionalities of each layer in OSI model.
13. Analyse different types of transmission impairments occurring in transmission media in detail.

OR

14. Explain in detail about guided and unguided media used for data transmission.
15. Discuss in detail about different line coding schemes.

OR

16. Identify the different techniques used for changing an analog signal to digital data.
17. Illustrate in detail about Time division multiplexing. With the help of neat sketch explain different schemes of Time division multiplexing.

OR

18. Examine the need of spread spectrum? Which are different spread spectrum techniques?
19. Discuss in detail about different channelization protocols.

OR

20. Demonstrate in detail about Bluetooth .List out its applications.

Syllabus

Module 1 (7 Hours)
Introduction to Data Communication- Components, Data Representation, Data Flow. Networks - Network Criteria, Physical structures, Physical Topology, Network Types- LAN, WAN, Switching -Internet -Network Models-OSI Model.
Module 2 (9 Hours)
Transmission media – Guided media – Twisted pair cable, coaxial cable, fiber optic cable, Unguided media - Radio waves, Microwaves, Infrared. Data and signals - Periodic analog signals-digital signals-transmission impairment - Attenuation, Distortion - Noise- different types of noise – Data rate limits-Noiseless channel, Noisy Channel, Performance
Module 3 (10 Hours)
Digital data transmission – Digital to Digital Conversion –Line Coding, Line Coding Schemes, Block coding, Transmission modes- Serial, Parallel, Synchronous, Asynchronous and Isochronous transmission. Encoding analog data into analog signals - AM, FM, PM. Encoding analog data into digital signals - PCM, DM – Keying Techniques - ASK, FSK, PSK, QAM
Module 4 (7 Hours)
Multiplexing- Frequency Division Multiplexing (FDM) – Time Division Multiplexing (TDM), Synchronous Time Division Multiplexing –Statistical time Division multiplexing Spread spectrum-The concept of spread spectrum – frequency hopping spread spectrum – direct sequence spread spectrum. Basic Principles of Switching-Circuit Switch Networks, Packet Switching, Structure of Switch
Module 5 (12 Hours)
Media Access Control – Random Access - Controlled Access-Channelization. Introduction to Wireless LAN-IEEE 802.11 –Bluetooth- WiMax, Cellular Telephony - 1G, 2G, 3G, 4G, 5G.

Text Books

1. Behrouz A Forouzan, “Data Communication and Networking”, McGraw Hill Education(india)Private limited,Fifth edition, 2013.

Reference Books

1. Stallings W., Data and Computer Communications, 8/e, Prentice Hall, 2007.

2. Tanenbaum A. S and D. Wetherall, Computer Networks, Pearson Education, 2013

3. Taub & Schilling, Principles of Communication Systems: Tata McGraw-Hill

4. Simon Haykin,Communication Systems: John Wiley & Sons. Pvt. Ltd

5. Das, Mullick & Chatterjee, Principles of Digital Communication: Wiley Eastern Ltd.

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Overview (7 Hours)**	
1.1	Introduction to Data Communication and networks	3
1.2	Network Models-OSI Model	4
2	**Transmission media (9 Hours)**	
2.1	Guided media and UnGuided media	2
2.2	Data and signals	3
2.3	Transmission impairments	2
2.4	Data rate limits and performance	2
3	**Digital data transmission (10 Hours)**	
3.1	Digital to digital Conversion	5
3.2	Transmission modes	1
3.3	Encoding analog data into analog signals	1
3.4	Encoding analog data into digital signals	2
3.5	Keying Techniques	1
4	**Bandwidth utilization: Multiplexing and Spectrum spreading, Switching (7 Hours)**	
4.1	Multiplexing	3
4.2	Spread spectrum	2
4.3	Basic Principles of switching	2
5	**Media Access Control and Wireless LANs (12 Hours)**	
5.1	Random Access	2
5.2	Channelization	1
5.3	Wireless LANs	5
5.4	Other Wireless networks	4

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT285	SOFTWARE ENGINEERING	VAC	3	1	0	4

Preamble: The syllabus is prepared with the view of preparing the Engineering Graduates capable of understanding essential concept of software engineering and software development process.

Prerequisite: Basics of programming

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category
CO1	**Summarize** different software development models	Level 2: Understand
CO2	**Identify** methods for requirement analysis, specification, design & testing	Level 3: Apply
CO3	**Explain** the software quality maintenance measures	Level 2: Understand
CO4	**Explain** the role of people in Software Engineering	Level 2: Understand
CO5	**Analyze** the risk factors and project management in Software Development	Level 3: Apply
CO6	**Illustrate** the legal and business aspects of Software Engineering	Level 2: Understand

Mapping of course outcomes with program outcomes

COs	PROGRAMME OUTCOMES (PO)											
	K3 PO1	4 PO2	5 PO3	5 PO4	6 PO5	3 PO6	2 PO7	3 PO8	3 PO9	3 PO10	3 PO11	3 PO12
CO1	1	2	3	-	3	2	-	2	3	1	2	1
CO2	2	3	3	1	3	-	-	1	2	2	3	1
CO3	1	1	-	-	3	1	-	1	-	1	-	1
CO4	-	1	-	1	-	1	-	1	3	3	2	1
CO5	3	3	-	1	2	-	-	-	-	-	2	1
CO6	-	-	-	-	-	3	-	3	-	-	2	1

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	20	20
Apply	20	20	70
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Test (2 numbers) : 25 marks

Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain importance of different phases of software development
2. List out the advantages and disadvantages of Spiral model
3. Illustrate the advantages of Agile model over other models.
4. Describe waterfall model in detail.
5. Explain Agile model in detail

Course Outcome 2 (CO2):

1. List out the requirement specification methods
2. Describe different architecture styles
3. Identify different UML diagrams to do software design
4. Explain unit, integration and system testing in detail.
5. Differentiate different system models.
6. Develop an SRS for a MobileApp that does caller identification.

Course Outcome 3 (CO3):

1. Write short note on reliability metrics.

2. Illustrate the steps in software configuration management.
3. Explain the functioning of DevOps.

Course Outcome 4 (CO4):

1. Analyze the role of users and staff in problem solving.
2. Identify the people factors in human driven software engineering.
3. Discuss the responsibilities of each personnel in a project.

Course Outcome 5 (CO5):

1. Analyze the risk factors in a MobileApp project
2. Describe the phases of software project management
3. Examine different types of documents generated during the software development.
4. Explain the software maintenance methods.
5. Discuss the responsibilities of a project manager.

Course Outcome 6 (CO6):

1. Describe intellectual property in the context of software.
2. Summarize the data privacy policies in software engineering
3. Illustrate different business models with examples.

Model Question Paper

PART A **(10*3=30)**

(Each Question carries 3 Marks)

1. List out the advantages and disadvantages of Spiral model
2. Examine the importance of different phases of software development
3. Classify the software requirements.
4. Compare and contrast decision tree and decision table.
5. Summarize the objectives of testing.
6. Differentiate cohesion and coupling in software design.
7. Write short note on reliability metrics.
8. Analyze the role of users in problem definition.
9. Analyze the risk factors in a MobileApp project
10. Summarize the data privacy policies in software engineering

PART B **(5*14=70)**

11. Describe each phase is software development life cycle in detail.

OR

12. Explain Agile model in detail
13. Discuss the formal methods of requirement specification

OR

14. Demonstrate the format of an SRS with a suitable example.
15. Illustrate different architecture styles with neat diagrams

OR

16. Identify different UML diagrams to do software design
17. Illustrate the steps in software configuration management

OR

18. Discuss the responsibilities of each personnel in a project
19. Describe the phases of software project management

OR

20. Illustrate different business models with examples.

Syllabus

Module 1: Introduction to Software Engineering
Software Engineering -Introduction and Overview- Software Process Models.
Module 2: Feasibility Study, Requirements Analysis & specification
Feasibility Study. Requirements Analysis & specification, Modeling Techniques.SRS
Module 3: Software Design, Software Testing
Software design, System Models, Design Methods. System Architecture, Object Oriented Design. Software Testing -Testing Documentation and Help Facilities.
Module 4: Software Reliability, SCM, People and Software Engineering
Software Reliability, Testing and bug fixing Tools, Software Configuration Management, Software Project Management, People and Software Engineering.
Module 5: Software documentation, delivery & maintenance, SPM, legal & business aspects
Software documentation, delivery & maintenance, Software Project Management, Legal Aspects of Software Engineering, Business Aspects of Software Engineering.

Text Books

T1. Roger S. Pressman, Software Engineering: A practitioner's approach, 8th Edition (Indian Edition), McGraw Hill. 2019

T2. Rajib Mall, Fundamentals of Software Engineering, 5th Edition, Prentice Hall India. 2018

Reference Books

R1. Pankaj Jalote, An integrated approach to Software Engineering, 3rd Edition, Springer/Narosa.

R2. Ian Sommerville, Software Engineering, 10th Edition, Addison-Wesley

R3. Sunitha EV, Sarath KS, Software Development Life Cycle: Theory vs Practice, Jyothis Publishers, 2019.

R4. Pfleeger, Atlee, Software Engineering Theory and Practice, 4 edition, 2009, Pearson.

R5. Grady Booch, Object-Oriented Analysis and Design with Applications, 3rd Edition, Pearson.

Course Contents and Lecture Schedule

Sl.No	**Module 1: Introduction to Software Engineering**	**7hrs**
1.1	**Introduction to Software Engineering**- overview of the software development process, importance of each step in software development, Cases of software project failures of major companies.	1
1.2	**Software Process Models**- Waterfall model - phases, pros and cons of the model, application (detailed description of each phase is required)	2
1.3	**Prototyping model** - phases, pros and cons of the model, application	1
1.4	**Spiral model** - phases, pros and cons of the model, application	1
1.5	**Rapid Software Development**, Agile model - phases, pros and cons of the model, application	2
	Module 2: Feasibility Study, Requirements Analysis & specification	**9 hrs**
2.1	**Feasibility Study** - methods, feasibility report	1
2.2	**Requirements Analysis & specification** - importance of requirements, types of requirements, Requirement Analysis, modeling and specification steps.	1
2.3	**Scenarios and Use Cases** - case study	1
2.4	**Informal Methods of Specification** - advantages and disadvantages **Formal Methods of Specification -** Axiomatic and Algebraic specifications	2
2.5	**Modeling Techniques** for Requirements Analysis and Definition - DFD – ER Diagrams – Decision tables – Decision Trees	3
2.6	**Software Requirement Specification** - format, importance, fit and gap analysis	1
	Module 3: Software Design, Software Testing	**12 hrs**
3.1	**Software design** – Cohesion and Coupling, Usability, UI/UX design, System Models: Data-flow models, Semantic data models, Object models. Design Methods- object oriented and function oriented.	3
3.2	**System Architecture** - Architectural Styles, Software Considerations of System Architectures.	1

3.3	**Object Oriented Design** – overview of UML diagrams, Tools and Techniques, design reuse.	2
3.4	Web App and Mobile App design – responsive design.	1
3.5	**Software Testing** - Objectives of testing, Testing Principles. Functional and Structural testing	3
3.6	Generation of test data - Test Plan - Unit testing – Integration testing – System testing. Testing GUIs, Test reporting, Testing Documentation and Help Facilities.	1
3.7	Testing Object-Oriented Applications, Web Apps and Mobile Apps	1
	Module 4: Software Reliability, SCM, People and Software Engineering	**9 hrs**
4.1	**Software Reliability** - Reliability metrics, The development process Reviews, Different aspects of reliability Programming techniques, Testing and bug fixing Tools, Performance testing.	3
4.2	**Software Configuration Management** – steps, features and tools, case study with DevOps.	2
4.3	**People and Software Engineering** - Software Development Staff and roles, The importance of people in problem solving process: The Role of Users in Problem definition;	2
4.4	**Human driven software engineering**; The people factor – Multidisciplinary aspects; The team factor; The customer factor.	2
	Module 5: Software documentation, delivery & maintenance, SPM, legal & business aspects	**8 hrs**
5.1	**Software documentation, delivery & maintenance**, Categories of Documentation, categories of software products, software maintenance methods.	2
5.2	**Software Project Management** - phases of Project Management, Project plan, Risk analysis.	2
5.3	**Legal Aspects of Software Engineering**- Contracts and licenses, Software Copyright, Software Patents, Trade Secrets and Non-Disclosure Agreements, Privacy.	2
5.4	**Business Aspects of Software Engineering** - Business Models. Emerging Trends in Software Engineering	2

SEMESTER -4

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT202	PRINCIPLES OF OBJECT ORIENTED TECHNIQUES	PCC	3	1	0	4

Preamble:

This course is intended to make the students capable of

1. Compare the capabilities of Object oriented and Procedure oriented programming languages.
2. Model the problem scenarios using object oriented concepts and UML.
3. Develop robust programs by optimally utilising the capabilities JAVA programming language.

Prerequisite: Programming Concepts

Course Outcomes: After the completion of the course the student will be able to

CO No	Course Outcomes	Bloom's Category
CO 1	Describe the specific capabilities of Object-Oriented paradigm compared to procedure oriented paradigm	Understand
CO 2	Explain the use of object oriented concepts to analyse the given problem.	Understand
CO 3	Describe the different inheritance features available in Java	Understand
CO 4	Construct robust programs using Exception Handling	Apply
CO 5	Construct applets utilising multithreading, event handling and Graphical User Interface, also model the problem scenarios using UML diagrams.	Apply

Mapping of course outcomes with program outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	3	3	-	2	-	-	-	-	-	-	1
CO 2	2	3	3	-	2	-	-	-	-	-	-	-
CO 3	2	3	3	-	2	-	-	-	-	-	-	-
CO 4	2	3	3	-	2	-	-	-	-	-	-	-
CO 5	2	3	3	-	2	-	-	-	-	-	-	-

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember			
Understand	25	25	50
Apply	25	25	50
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Compare and contrast the implementation of data abstraction in procedure oriented and object oriented language.
2. Explain how encapsulation helps in data security. Justify your answer by comparing the scenario in procedure oriented languages.
3. Describe TWO features of object oriented programming languages that promote code reuse.

Course Outcome 2 (CO2):

1. Describe about the statement “String is a primitive data type or not in C++.

2. Describe the use of ‘static’ functions in C and Java.

Course Outcome 3(CO3):

1. Explain the difference between the object oriented design concepts of generalisation and specialisation, and describe how these relate to the inheritance feature in object oriented programming languages.
2. Describe how is-a and has-a inter-class relationships may be implemented in object oriented programming, giving code examples to support your answer.
3. What is multiple inheritance? Discuss how multiple inheritance is implemented in Java.

Course Outcome 4(CO4):

1. Experiment with runtime and compile time errors. Would you rather have an error discovered at run time or compile time?
2. Experiment with out of bound exception with example code.
3. Build java programs using following constructs.

   a) try {}

   b) catch {}

   c) throw()

Course Outcome 5(CO5):

1. Experiment with multithreaded applets.

2. Experiment with inter-process communication mechanism in java.

3. Design a class diagram of the following System: Vending Machine. A vending machine sells small, packaged, ready to eat items (chocolate bars, cookies, candies, etc.). Each item has a price and a name. A customer can buy an item, using a smart card (issued by the vending machine company) to pay for it. No other payment forms (i.e. cash, credit card) are allowed. The smart card records on it the amount of money available. The functions supported by the system are: Sell an item(choose from a list of items, pay item, distribute item)Recharge the machine Set up the machine (define items sold and price of items)Monitor the machine (number of items sold, number of items sold per type, total revenue)The

system can be used by a customer, a maintenance employee (who recharges items in the machines), an administrator (who sets up the machine).

Model Question paper

Course Code: ITT202

Course Name: PRINCIPLES OF OBJECT ORIENTED TECHNIQUES

Max.Marks:100 **Duration: 3 Hours**

PART A

Answer all Questions. Each question carries 3 Marks

1. Explain the use of 'static' functions in C and Java.
2. Comment on the statement "String is a primitive data type or not in C++.
3. Explain the use of 'this' keyword in Java.
4. Write a short note on implementation of "write once, run anywhere" concept in Java
5. Explain encapsulation using class in java.
6. Write short note on inner calss in Java.
7. Explain the use of dynamic method dispatch.
8. Differentiate between 'throw', 'throws', and 'Throwable'.
9. Explain event handling model with diagram.
10. Write short note on sand box security model for applets.

Part B

Answer any one Question from each module. Each question carries 14 Marks

11.a) Compare and contrast the implementation of data abstraction in procedure oriented and object oriented language. 7Marks

b) Discuss how encapsulation helps in data security. Justify your answer by comparing the scenario in procedure oriented languages. 7 Marks

OR

12. a) Write a short note on garbage collection in java. 6 Marks

b) Describe with example, TWO features of object oriented programming languages that promote code reuse. 8 Marks

13. a) Compare and contrast overriding and overloading with examples.

6 Marks

b) With an example, discuss the use of passing objects as parameters.

8 Marks

OR

14.a) Discuss the uses of 'static' keyword with example scenarios. 7 marks

b) Examine the use of 'final' keyword in the context of access control.

7Marks

15.a) Explain the difference between the object oriented design concepts of generalisation and specialisation, and describe how these relate to the inheritance feature in object oriented programming languages. 6 marks

b) Describe how is-a and has-a inter-class relationships may be implemented in object oriented programming, giving code examples to support your answer.

8 Marks

OR

16.a) What is multiple inheritance? Discuss how multiple inheritance is implemented in Java. 7 marks

b) Compare and contrast the usage of abstract class and interface in Java. Give examples of each. 7 Marks

17.a) Discuss the difference between runtime and compile time errors. Would you rather have an error discovered at run time or compile time? 8 Marks

b) Explain out of bound exception with example code. 6 marks

OR

18. a) Explain the use of following constructs in Java with example.

i)try {}

ii)catch {}

iii)throw() 6 Marks

b) Elaborate on the interprocess communication mechanism in java.

8 Marks

19. a) Discuss how to implement a multithreaded applet with an example.

6 Marks

b) Explain with an example the event model in Java. 8 Marks

OR

20. a) Differentiate between static and dynamic models in UML. 4 Marks

b) Draw a class diagram of the following System: Vending Machine. A vending machine sells small, packaged, ready to eat items (chocolate bars, cookies, candies, etc.). Each item has a price and a name. A customer can buy an item, using a smart card (issued by the vending machine company) to pay for it. No other payment forms (i.e. cash, credit card) are allowed. The smart card records on it the amount of money available .The functions supported by the system are:Sell an item(choose from a list of items, pay item, distribute item)Recharge the machine Set up the machine (define items sold and price of items)Monitor the machine (number of items sold, number of items sold per type, total revenue)The system can be used by a customer, a maintenance employee (who recharges items in the machines), an administrator (who sets up the machine).

10 Marks

Syllabus

Module 1	No. of Lectures
Object-Oriented Programming vs Procedure-Oriented Programming, Procedural Languages - The Object-Oriented Approach - Characteristics of Object-Oriented Languages Objects –Classes – Inheritance – Reusability - Creating New Data Types - Polymorphism and Overloading ,Object oriented concepts in Java -Java Overview: Java virtual machine, data types, operators, control statements, Classes fundamentals, objects, methods, constructors, this keyword, Garbage collection	10 hours
Module 2	
Overloading Methods, Overloading Constructors, Using Objects as Parameters, Call by value and Call by reference, Acess control, use of static and final keywords, Nested and Inner classes	8 hours
Module 3	
Derived Class and Base Class, Usage of super keyword, Creating a Multilevel Hierarchy, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, Definition and appliction of Packges and Interfaces	8 hours
Module 4	
Fundamentals of exception handling, Exception Types, Using try and catch, throw, throws, finally, Java's Built-in Exceptions, Creating Exception subclasses, the Java Thread Model, Creating a Thread, Creating Multiple Threads, Thread Priorities, Synchronization, Interthread	9 hours

Communication	
Module 5	
Event Handling-delegation event model, event classes, sources, listeners. String class - basics. Applet basics and methods, AWT- working with frames, graphics, color, font. AWT Control fundamentals. Swing overview, Introduction to Object Oriented Modelling ,Unified Modeling Language**,** UML class diagram, Use-case diagram, Familiarisation of UML tools, Case study	10 hours

Text Books

1. Herbert Schildt, Java: The Complete Reference, 8/e, Tata McGraw Hill, 2011.

2. Bahrami A., Object Oriented Systems Development using the Unified Modeling

Language, McGraw Hill, 1999.

Reference Books

1. Flanagan D., Java in A Nutshell, 5/e, O'Reilly, 2005.

2. Sierra K., Head First Java, 2/e, O'Reilly, 2005.

3. Balagurusamy E., Programming JAVA a Primer, 5/e, McGraw Hill, 2014.

4. Barclay K., J. Savage, Object Oriented Design with UML and Java, Elsevier, 2004.

5. James Rumbaugh.,Unified Modeling Language Reference Manual,Addison-Wesley Professional,2005

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Object Oriented concepts**	**10 Hours**
1.1	Object-Oriented Programming vs Procedure-orientated Programming, Procedural Languages - The Object-Oriented Approach - Characteristics of Object-Oriented Languages	3 hours
1.2	Objects –Classes – Inheritance – Reusability - Creating New Data Types - Polymorphism and Overloading	3 hours
1.3	Object oriented concepts in Java -Java Overview: Java virtual machine, data types, operators, control statements, Classes fundamentals, objects, methods, constructors, this keyword, Garbage collection	4 hours
2	**Method overloading**	**8 Hours**

2.1	Overloading Methods, Overloading Constructors, Using Objects as Parameters	4 hours
2.2	Access control, use of static and final keywords, Nested and Inner classes	4 hours
3	**Inheritance**	**8 Hours**
3.1	Derived Class and Base Class, Usage of super keyword, Creating a Multilevel Hierarchy, Method Overriding	4 hours
3.2	Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, Definition and application of Packages and Interfaces	4 hours
4	**Exception handling and Multithreaded Programming**	**9 Hours**
4.1	Fundamentals of exception handling, Exception Types, Using try and catch, throw, throws, finally, Java's Built-in Exceptions, Creating Exception subclasses.	4 hours
4.2	The Java Thread Model, Creating a Thread, Creating Multiple Threads, Thread Priorities, Synchronization, Interthread Communication	5 hours
5	**Event Handling, AWT and UML**	**10 Hours**
5.1	Event Handling-delegation event model, event classes, sources, listeners. String class – basics. Applet basics and methods	3 hours
5.2	AWT- working with frames, graphics, color, font. AWT Control fundamentals. Swing overview	3 hours
5.3	Introduction to Object Oriented Modelling-Unified Modelling Language, UML class diagram, Use-case diagram, Familiarisation of UML tools, Case study	4 hours

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT204	COMPUTER ORGANIZATION	PCC	3	1	0	4

Preamble

This syllabus has been prepared to meet the following objectives.

1. To impart an understanding of the internal organization and operations of a computer.
2. To introduce the concepts of processor logic design.
3. To introduce the concept of pipe-lining and its hazards.
4. To understand and analyze various issues related to memory hierarchy.
5. To introduce the various modes of data transfer between CPU and I/O devices.

Prerequisite: ITT201 Digital System Design

Course Outcomes: After the completion of the course the student will be able to

CO No	Course Outcome(CO)	Bloom's Category
CO 1	Describe the basic organization of computer and different instruction formats and addressing modes.	Understand
CO 2	Analyze the basic operations and sequencing of control signals	Analyze
CO 3	Represent the design of registers and arithmetic logic unit	Understand
CO 4	Examine the concept of pipe-lining and various hazards associated with it	Analyze
CO 5	Compare the performance of memory systems like cache and DRAM and Select appropriate interfacing standards for I/O devices.	Analyze

Mapping of course outcomes with program outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	-	-	-	-	-	-	-	-	1	1
CO2	3	3	3	2	-	-	-	-	-	-	1	1
CO3	3	2	3	-	-	-	-	-	-	-	1	1
CO4	3	3	3	3	2	-	-	-	-	-	1	2
CO5	3	3	3	3	1	-	-	-	-	-	1	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Test		End-Semester Examination
	1	2	
Remember	**12**	**8**	**16**
Understand	**10**	**12**	**24**
Apply	**20**	**20**	**40**
Analyse	**8**	**10**	**20**
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):
1.Explain the importance of different addressing modes in computer architecture with suitable example
2.How is the operation X = (A + B) * (C + D) / (E+F) is performed using:
a) Three address instruction
b) Two address instruction
c) One address instruction

Course Outcome 2 (CO2):

1. Enumerate the sequence of actions involved in executing an unconditional branch instruction.
2.Write down the sequence of actions needed to fetch and execute the instruction:
   Store R6, X(R8).

Course Outcome 3 (CO3):

1.Draw the block diagram for the hardware that implements the following statement x + yz: AR ← AR + BR where AR and BR are two n-bit registers and x, y, and z are control variables. Include the logic gates for the control function. (The symbol + designates an OR operation in a control or Boolean function and an arithmetic plus in a micro operation.)
2. Illustrate the difference in performance of an Arithmetic Right Shifter & a Logical Right Shifter

Course Outcome 4 (CO4):

1.A5-stage pipelined processor has Instruction Fetch(IF), Instruction Decode (ID), Operand Fetch (OF), Perform Operation (PO) and Write Operand (WO) stages. The IF, ID, OF and WO stages take 1 clock cycle each for any instruction. The PO stage takes 1 clock cycle for ADD and SUB instructions, 3 clock cycles for MUL instruction, and 6 clock cycles for DIV instruction respectively. Operand forwarding is used in the pipeline. What is the number of clock cycles needed to execute the following sequence of instructions?

Instruction	Meaning of instruction
I1 :MUL R2 ,R0 ,R1	R2 = R0 *R1
I1 :DIV R5 ,R3 ,R4	R5 = R3/R4
I2 :ADD R2 ,R5 ,R2	R2 = R5+R2
I3 :SUB R5 ,R2 ,R6	R5 = R2-R6

2. The instruction pipeline of a RISC processor has the following stages: Instruction Fetch (IF), Instruction Decode (ID), Operand Fetch (OF), Perform Operation (PO) and Writeback (WB), The IF, ID, OF and WB stages take 1 clock cycle each for every instruction. Consider a sequence of 100 instructions. In the PO stage, 40 instructions take 3 clock cycles each, 35 instructions take 2 clock cycles each, and the remaining 25 instructions take 1 clock cycle each. Assume that there are no data hazards and no control hazards. How many clock cycles are required for completion of execution of the sequence of instruction?

Course Outcome 5 (CO5):

1. A computer has a 256 KByte, 4-way set associative, write-back data cache with block size of 32 Bytes. The processor sends 32 bit addresses to the cache controller. Each cache tag directory entry contains, in addition to address tag, 2 valid bits, 1 modified bit and 1 replacement bit. How many bits are there in the tag,set and word field of an address?
2. Discuss DRAM scheduling policies.

Course Outcome 6 (CO6):

1. What is the basic advantage of using interrupt initiated data transfer over transfer under program control without an interrupt? What is asynchronous data transfer? Explain in detail.
2. Explain the working of Universal Serial Bus (USB).

Model Question Paper

PART A **(10*3=30)**

(Each question carries 3 Marks)

1. What are fundamental phases of the instruction cycle?
2. The register R1 = 12, and R2= 13. The instruction ADD R1, R2 is in memory location 2000H. After the execution of the instruction, what will be the value of PC, MAR, IR and R1?.
3. What do you meant by logic micro operations?
4. Design a 4bit combination logic shifter.
5. “Increasing the number of pipeline stages will decrease the execution time of the program”. True or False? Justify your answer.
6. What is operand forwarding? What is its significance?
7. For a 16KB, 4-way associative cache with block size 16 bytes, what is the number of tag bits per block if the physical address capacity is 16MB?
8. List the advantages of memory interleaving
9. Compare Polling and Vectored Interrupts.
10. What is DMA? What do you meant by Burst mode?

PART B **(5*14=70)**

(Each full question carries 14 marks)

11.a) Discuss the sequencing of control signals for the following instructions.

i) Load R1,10(R2) ii) Add R1, R2 (8 marks)

b) Compare and contrast memory mapped IO over programmed IO. (6 marks)

OR

12.a) Illustrate with example, explain the different types of addressing modes in a RISC processor. (9 marks)

b) Discuss how stack used for subroutine call. (5 marks)

13.a) Design a 4 bit arithmetic unit with two selection variables s0 and s1 and two n-bit data inputs A&B and input carry Cin

s1	s2	Cin=0	Cin=1	
0	0	F=A	F=A+1	
0	1	F=A+B	F=A+B+1	
1	0	F=A+B’	F=A+B’+1	
1	1	F=A-1	F=A	

(9 marks)

b) Explain the design of an accumulator. (5 marks)

OR

14. a) Design an adder/subtractor circuit with one selection variable s and two inputsA and B. When s=0, the circuit performs A+B and when s=1 it performs A-B, by taking 2's complement of B. (9marks)

b) Explain the design of status register. (5 marks)

15.a) Consider an instruction pipeline with four stages with the stage delays 5 nsec, 6 nsec, 11 nsec, and 8 nsec respectively. The delay of an inter-stage register stage of the pipeline is 1 nsec. What is the approximate speedup of the pipeline in the steady state underideal conditions as compared to the corresponding non-pipelined implementation? (5 marks)

b) Discuss structural hazards and control hazards with examples (9 marks)

OR

16. a) A 5-stage pipelined processor has the stages: Instruction Fetch (IF), Instruction Decode (ID), Operand Fetch (OF), Execute (EX) and Write Operand (WO). The IF, ID, OF, and WO stages take 1 clock cycle each for any instruction. The EX stage takes 1 clock cycle for ADD and SUB instructions, 3 clock cycles for MUL instruction, and 6 clock cycles for DIV instruction. Operand forwarding is used in the pipeline (for data dependency, OF stage of the dependent instruction can be executed only after the previous instruction completes EX). What is the number of clock cycles needed to execute the following sequence of instructions?

MUL R2,R10,R1

DIV R5,R3,R4

ADD R2,R5,R2

SUB R5,R2,R6 (7 marks)

b) Discuss various types data hazards in a RISC Instruction pipeline with appropriate examples. (7 marks)

17. a) Consider an application running on a multiprocessor system that takes 600 cycles, (during which processors are stalled), to handle a local cache miss leading to referencing a remote memory. The CPI for all references that hit in cache is 1 cycle. If 0.2% of cache access result in a local miss, how much faster will the system run if it has a perfect cache that never miss. (5 marks)

b) Discuss organization of DRAM in detail. (9 marks)

OR

18.a) Discuss open page and closed page row buffer management policy in DRAM Controller (9 marks)

b) Given a cell array of 8K(8192), with Clock cycle=4 and Clock Rate=133MHZ. In DRAM, the period for refreshing all rows is 16ms whereas 64ms in SDRAM. Find out the Refresh Overhead of SDRAM when compared to DRAM (5 marks)

19.a) Discus different types of interrupt handling methods (7 marks)

b) Explain the working of SCSI. (7 marks)

OR

20.a)Discuss various bus arbitration methods. (7 marks)

b) Explain the working of PCI. (7 marks)

Syllabus

Module 1 (10 hours)
Basic Structure and Operation of Computers – functional units –operational concepts – memory operations – addressing modes – instruction sequencing – basic I/O – subroutine calls – execution of a complete instruction – sequencing of control signals.
Module 2 (8 hours)
Processor Logic Design and Organization – register transfer logic – micro operations – conditional control statements. Design of arithmetic unit, logic unit, ALU and shifter – Accumulator.
Module 3 (9 hours)
RISC – RISC instruction set – pipelining – hazards and mitigation.
Module 4 (11 hours)
Memory – cache memory, mapping and performance improvement. DRAM organization. Memory controllers-scheduling
Module 5 (7 hours)
Peripheral Subsystem – I/O organization – interrupts – DMA – bus arbitration – standard I/O interfaces.

Text Books:

1. Patterson D.A. and J. L. Hennessey, Computer Organization and Design, 5/e, Morgan Kauffmann Publishers, 2013.
2. Hamacher C., Z. Vranesic and S. Zaky, Computer Organization,5/e, McGraw Hill,2011.
3. M. Morris Mano, Computer System Architecture, 3/e, Pearson Education, 2007.
4. Bruce Jacob, David T. Wang, and Spencer Ng, Memory Systems: Cache, DRAM, Disk, 1/e Morgan Kauffmann Publishers, 2007.

References:

1. William Stallings, Computer Organization and Architecture: Designing forPerformance, Pearson, 9/e, 2013.
2. Computer Architecture: Pipelined and Parallel Processor Design M.J. Flynn Published byNarosa Publishing House, 2012
3. Rajaraman V. and T. Radhakrishnan, Computer Organization and Architecture,Prentice Hall, 2011.
4. Messmer H. P., The Indispensable PC Hardware Book, 4/e, Addison-Wesley, 2001

Course Content & Lecture Schedule

	Module 1: Fundamentals of Computer Organization	**10 hours**
1.1	Basic Structure of computers–functional units –basic operational concepts –bus structures –software.	**2 hours**
1.2	Memory locations and addresses –memory operations – instructions and instruction sequencing – addressing modes – ARM Example (programs not required).	**4 hours**
1.3	Basic I/O operations –stacks, subroutine calls. Basic processing unit – fundamental concepts –instruction cycle - execution of a complete instruction – multiple-bus organization –sequencing of control signals.	**4 hours**
	Module 2:Processor Logic Design	**8 hours**
2.1	Register transfer logic –inter register transfer – arithmetic, logic and shift micro operations –conditional control statements.	**4 hours**
2.2	Design of arithmetic unit, logic unit, arithmetic logic unit and shifter – status register –processor unit –design of accumulator.	**4 hours**
	Module 3: RISC Instruction Pipelining	**9 hours**
3.1	Introduction to RISC instruction set, load store architecture	**3 hours**
3.2	Overview of pipelining, pipelined datapath and control	**2 hours**
3.3	Pipeline hazards, hazard mitigation techniques.	**4 hours**
	Module 4: Memory system : Cache & DRAM	**11 hours**
4.1	Introduction to cache memory, cache mapping, block replacement techniques, measuring and improving cache performance .	**4 hours**
4.2	Introduction to DRAM system, DRAM organization-Memory interleaving.	**2 hours**
4.3	Memory controllers, Address mapping, DRAM Scheduling policies, Row Buffer management policies- DRAM Refreshing	**5 hours**

	Module 5: Peripheral Subsystem	**7 hours**
5.1	I/O organization: accessing of I/O devices –interrupts	**2 hours**
5.2	Direct memory access –buses –bus arbitration	**2 hours**
5.3	Interface circuits –standard I/O interfaces (PCI, SCSI, USB)	**3 hours**

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT206	Database Management Systems	PCC	3	1	0	4

Preamble: Database Management Systems course is intended to deliver students the elementary concepts of a database management system and equips them to design and implement a database application built over those concepts. It also introduces to them advanced level areas like transaction processing, concurrency control and recovery management. The current trend, unstructured data - NoSQL is unveiled too.

Prerequisite: NIL

Course Outcome (CO): After completion of the course, the student will be able to

CO No.	CO	Bloom's Category
CO1	Describe the fundamental concepts of databases.	Understand
CO2	Construct an Entity-Relationship (ER) model and transform to relational schema.	Apply
CO3	Develop queries for relational database in the context of practical applications.	Apply
CO4	Model and design relational databases following the design principles.	Apply
CO5	Describe the concepts of control and recovery techniques in transaction processing and NoSQL database.	Understand

Mapping of Course Outcomes with Program Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	1	3	-	-	-	-	-	-	-	-	-
CO2	2	2	3	-	-	-	-	-	-	-	-	-
CO3	1	1	2	2	3	-	-	-	-	-	-	2
CO4	2	2	3	2	-	-	-	-	-	-	-	-
CO5	2	2	3	-	-	-	-	-	-	-	-	-

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests(Marks)		End Semester Examination (Marks)
	1	2	
Remember	5		30
Understand	30	20	30
Apply	15	30	40

Mark Distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Test (2 numbers) : 25 marks

Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

CO1: Explain 3-Schema architecture of database system with the help of a neat diagram.

CO2: Construct an ER diagram for a college database given the following statements:

- A college contains many departments
- Each department can offer any number of courses
- Many instructors can work in a department
- An instructor can work only in one department
- For each department there is a Head
- An instructor can be head of only one department

- Each instructor can take any number of courses
- A course can be taken by only one instructor
- A student can enroll for any number of courses
- Each course can have any number of students

CO3: Build SQL queries for each of the following relation schema given below:

employee (<u>employee-name</u>, street, city)
works (<u>employee-name, company-name</u>, salary)
company (<u>company-name</u>, city)
manages (<u>employee-name, manager-name</u>)

a. Find the names, street addresses, and cities of residence of all employees who work for First Bank Corporation and earn more than $10,000.
b. Find all employees in the database who do not work for First Bank Corporation.
c. Find all employees in the database who earn more than each employee of Small Bank Corporation.
d. Find the company that has the most employees.
e. Find those companies whose employees earn a higher salary, on average, than the average salary at First Bank Corporation.

CO4: Experiment with the table shown below:

branchNo	*branchAddress*	*telNos*
B001	8 Jefferson Way, Portland, OR 97201	503-555-3618, 503-555-2727, 503-555-6534
B002	City Center Plaza, Seattle, WA 98122	206-555-6756, 206-555-8836
B003	14 – 8th Avenue, New York, NY 10012	212-371-3000
B004	16 – 14th Avenue, Seattle, WA 98128	206-555-3131, 206-555-4112

a. Why is this table not in 1NF?
b. Describe and illustrate the process of normalizing the data shown in this table to third normal form (3NF).
c. Identify the primary, alternate and foreign keys in your 3NF relations.

CO5: Consider the following four schedules due to three transactions (indicated by the subscript) using read and write on a data item x, denoted by r(x) and w(x) respectively. Check which one of them is conflict serializable.

a. r1(x); r2(x); w1(x); r3(x); w2(x)

b. r2(x); r1(x); w2(x); r3(x); w1(x)

c. r3(x); r2(x); r1(x); w2(x); w1(x)

d. r2(x); w2(x); r3(x); r1(x); w1(x)

Model Question paper

Course Code: ITT 206

Course Name: Database Management Systems

Max.Marks:100 **Duration: 3 Hours**

Part-A
(Answer all questions. Each question carries 3 marks)

1. List three significant differences between a file-processing system and a DBMS.
2. List three reasons why database systems support data manipulation using a declarative query language such as SQL, instead of just providing a library of C or C++ functions to carry out data manipulation.
3. Why are duplicate tuples not allowed in a relation?
4. What is union compatibility? Why do UNION, INTERSECTION, and DIFFERENCE operations require that the relations on which they are applied be union compatible?
5. What is done when INSERT operation violates one or more constraints?
6. What are assertions? How do they differ from triggers?
7. Let R(A,B,C,D,E,P,G) be a relational schema in which the following FDs hold: {AB->CD, DE->P, C->E, P->C, B->G}. What is the highest normal form the relation schema R is in?
8. Why are Armstrong's axioms considered sound and complete?
9. What are the ACID properties for data integrity in DBMS? Explain each of them.
10. Discuss about the lock compatibility matrix.

Part -B

(Answer one question from each module. Each question carries 14 marks)

Module -I

11.

a. Construct an E-R diagram for a hospital with a set of patients and a set of medical doctors. Associate with each patient a log of the various tests and examinations conducted. Make suitable assumptions that are valid. (8 marks)

b. Explain the 3-schema architecture for database systems with a diagram. (6 marks)

12.

a. Construct an E-R diagram for a car insurance company whose customers own one or more cars each. Each car has associated with it zero to any number of recorded accidents. Each insurance policy covers one or more cars and has one or more premium payments associated with it. Each payment is for a particular period of time, and has an associated due date, and the date when the payment was received. You can make suitable and valid assumptions. (8 marks)

b. What are the different types of database end users? Discuss the main activities of each. (6 marks)

Module -II

13.

a. Given below is the schema of a database that keeps track of student enrollment in courses and the books adopted for each course:

STUDENT(ssn, name, major, bdate)

COURSE(course#, cname, dept)

ENROLL(ssn, course#, quarter, grade)

BOOK_ADOPTION(course#, quarter, book_isbn)

TEXT(book_isbn, book_title, publisher, author)

Using relational algebra specify the following queries:

- List the number of courses taken by all students named John Smith in Winter 2009 (i.e., Quarter=W09).
- Produce a list of textbooks (include Course#, Book_isbn, Book_title) for courses offered by the 'IT' department that have used more than two books.
- List any department that has all its adopted books published by 'Pearson Publishing'.

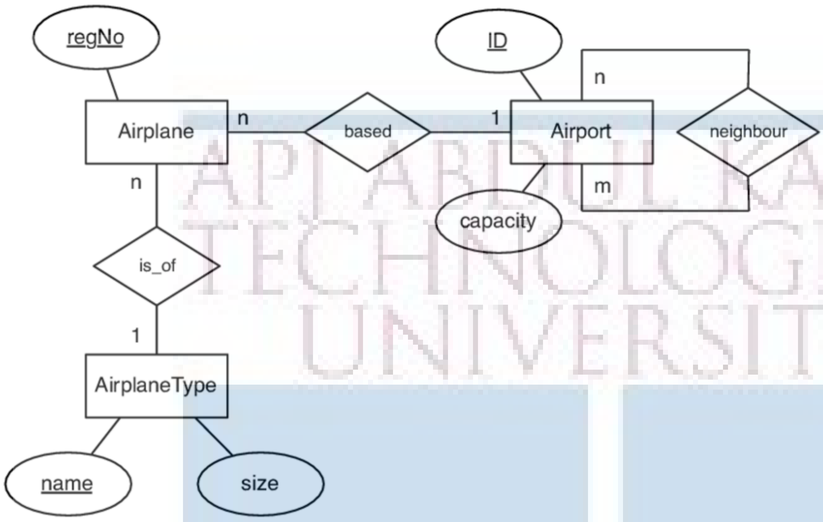
(9 marks)

b. What are the possible approaches for mapping binary 1:1 relations in an ER diagram?

(5 marks)

14.

a. Convert the ER model below into its equivalent relational schema: (5 marks)



b. Consider the following relational schema for a library:

MEMBER (memb_no, name, dob)

BOOKS (isbn, title, authors, publisher)

BORROWED (memb_no, isbn, date)

Write the following queries in relational algebra:

- Find the names of members who have borrowed any book published by "McGraw-Hill".
- Find the name and membership number of members who have borrowed more than five different books published by "MorganKaufmann".
- For each publisher, find the name and membership number of members who have borrowed more than five books of that publisher.

(9 marks)

Module-III

15.

a. For the EMPLOYEE schema given below:

EMPLOYEE (employee_name, street, city)

WORKS (employee_name, company_name, salary)

COMPANY (company_name, city)

MANAGES (employee_name, manager_name)

write SQL queries for the following:

- Find the names and cities of residence of all employees who work for "Gramin Bank Corporation".

- Find all employees in the database who earn more than each employee of “Co-operative Bank Corporation”.
- Assume that the companies may be located in several cities. Find all companies located in every city in which “Kerala Bank Corporation” is located.
- Find the company that has the most employees. (9 marks)

b. Why does SQL not automatically eliminate duplicate tuples in the results of its queries? (5 marks)

16.

a. For the database schema given below:

STUDENT (name, stud_no, class, major)

COURSE (course_name, course_no, credit_hours, department)

SECTION (section_identifier, course_no, semester, year, instructor)

GRADE_REPORT (stud_No, section_identifier, grade)

PREREQUISITE (course_no, prerequsitie_no)

write SQL queries for the following:

- Insert a new student, <‘Nikhila, 25, 1,‘Math’>, in the database.
- Change the class of student ‘Anirudh’ to 2.
- Insert a new course, <‘Knowledge Engineering’, ‘IT4390’, 3, ‘IT’>.
- Delete the record for the student whose name is ‘Kripa’ and whose student number is 17. (9 marks)

b. What is a correlated nested query in SQL? Give an example. (5 marks)

Module -IV

17.

a. Determine if the FD sets

$F = \{A \rightarrow C, AC \rightarrow D, E \rightarrow AD, E \rightarrow H\}$ and

$G = \{A \rightarrow CD, E \rightarrow AH\}$ are functionally equivalent. (7 marks)

b. Illustrate the following problems with suitable examples:

- Generation of spurious tuples
- Type-2 insertion anomaly

(7 marks)

18.

a. Using Ullman’s algorithm, check whether the relation schema R(A,B,C,D,E) decomposed into R1(A,D), R2(A,B), R3(B,E), R4(C,D,E) and R5(A,E) and the FD set

{A -> C

B -> C

C -> D

DE->C

CE -> A}

is lossy or lossless. (7 marks)

b. Given an FD set F= {A->BC, B->C, A->B, AB->C}. Find its minimal cover. (7 marks)

Module -V

19.

a. With a diagram, discuss the various states of a transaction. (4 marks)

b. What is log based recovery? Explain deferred database modification. (10 marks)

20.

a. Explain the two-phase locking protocol. How does it implement lock conversions to assure serializability? (10 marks)

b. What are checkpoints? How are they implemented? (4 marks)

Syllabus

MODULE 1: INTRODUCTION (9 HOURS)
Fundamentals of Database Management Systems (DBMS),Database System Concepts and Architecture, Entity-Relationship Model, ER Diagrams
MODULE 2: RELATIONAL MODEL (8 HOURS)
Relational Model Concepts, Transformation of ER diagram to Relational Schema, Relational Algebra Operations
MODULE 3: STRUCTURED QUERY LANGUAGE (SQL) (11 HOURS)
SQL Overview, Data Manipulation Language (DML), Advanced DML
Module 4: DATABASE DESIGN (7 HOURS)
Database Design Guidelines, Normalization using Functional Dependencies
Module 5: TRANSACTION PROCESSING AND INTRODUCTION TO NoSQL (10 HOURS)
Transaction Processing Concepts, Characterizing Schedules, Concurrency Control Techniques, Recovery Techniques, Introduction to NoSQL Databases

Text Books

1. Elmasri R. and S. Navathe, *Database Systems: Models, Languages, Design and Application Programming*, Pearson Education, 2013.

2. Silberschatz A., H. F. Korth and S. Sudarshan, *Database System Concepts*, 6/e, McGraw Hill, 2011.

Reference Books

1. C.J.Date, A.Kannan, S.Swamynathan, ―An Introduction to Database Systems, Eighth Edition, Pearson Education, 2006.
2. Raghu Ramakrishnan, ―Database Management Systems, Fourth Edition, McGraw-Hill College Publications, 2015.
3. G.K.Gupta,"Database Management Systems", Tata McGraw Hill, 2011.
4. Xun (Brian) Wu, SudarshanKadambi, DevramKandhare, Aaron Ploetz, Seven NoSQL Databases in a Week:1st Edition, Kindle Edition.

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**Introduction**	**9**
1.1	**Fundamentals of Database Management Systems**: Characteristics of DBMS, Database Users, Advantages of using DBMS.	2
1.2	**Database System Concepts and Architecture:** Data Models, Schemas, Instances and Database State. Three-Schema Architecture, Data Independence, Database Languages and Interfaces.	3
1.3	**Entity-Relationship Model**: Basic concepts - Entity, Attributes and Keys, Relationship Sets, Degree of Relationship Types, Roles and Structural Constraints.	2
1.4	**ER Diagrams**: Naming Conventions and Design Issues.	2
2	**Relational Model**	**8**
2.1	**Relational Model Concepts**: Domains, Attributes, Tuples and Relations, Relational Model Constraints and Relational Database Schemas.	3
2.2	**Transformation of ER diagram to Relational Schema.**	2
2.3	**Relational Algebra Operations:** SELECT, PROJECT, RENAME, Set Theoretic Operations, JOIN and DIVISION.	3
3	**Structured Query Language**	**11**

3.1	**SQL Overview**: Basic Structure, Data Definition Language Commands – CREATE, DROP and ALTER, Arithmetic Operations.	3
3.2	**Data Manipulation Language**: DML Commands - INSERT, SELECT, DELETE and UPDATE, Nested Queries, Set Operations, Aggregate Functions and Grouping, JOIN Operations	4
3.3	**Advanced DML:** Complex Queries, Views, Stored Procedures, Handling Exceptions and Triggers.	4
4	**Database Design**	**7**
4.1	**Database Design Guidelines**: Anomalies in Database Design – Insertion, Deletion and Modification, Functional Dependency (FD) – Closures, Armstrong’s Axioms, Equivalence,Minimal Cover (proofs not required).	3
4.2	**Normalization using Functional Dependencies**: Normal Forms(NF) - INF, 2NF, 3NF and Boyce - Codd Normal Form, Lossless Join and Dependency Preserving Decompositions.	4
5	**Transaction Processing and Introduction to NoSQL**	**10**
5.1	**Transaction Processing Concepts**: Transaction Concepts, ACID Properties, Transaction States.	2
5.2	**Characterizing Schedules:** Based on Recoverability and Serializability.	2
5.3	**Concurrency Control Techniques**: Types of Locks, Lock Based Protocols-Two Phase Locking protocol, Timestamp Based Protocols, Deadlock and Starvation.	3
5.4	**Recovery Techniques**: Recovery Based on Deferred Update and Immediate Update, Shadow Paging.	2
5.5	**Introduction to NoSQL Databases**	1

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITL202	OBJECT ORIENTED TECHNIQUES LAB	PCC	0	0	3	2

Preamble:

This lab is intended to make the students capable of

Understanding the importance of Object Oriented Programming in designing the Software applications,

Implementing programs using Object oriented concepts of inheritance and polymorphism,

Analysing the given problem to design multithreaded programs,

Developing robust programs using exception handling features in Java.

Prerequisite: ITT202 PRINCIPLES OF OBJECT ORIENTED TECHNIQUES

Course Outcomes: After the completion of the course the student will be able to

CO No	Course Outcome(CO)	Bloom's Category
CO 1	**Solve** the given problem by applying Object oriented features and Java concepts.	Apply
CO 2	**Implement** the concept of method and constructor overloading	Apply
CO 3	**Implement** the concept of inheritance Apply	Apply
CO 4	**Use** the concept of multithreading and modify an existing program with proper exception handling	Apply
CO 5	**Build** Robust programs in JAVA using AWT and SWING	Apply

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	2	1	2	-	-	-	-	-	-	1
CO 2	2	2	2	-	2	-	-	-	-	-	-	1
CO 3	2	2	2	-	2	-	-	-	-	-	-	1
CO 4	2	3	3	1	2	-	-	-	-	-	-	2
CO 5	2	3	3	1	3	-	-	-	-	-	-	2

3/2/1: High/Medium/Low

Assessment Pattern

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	75	75	2.5 hours

Continuous Internal Evaluation Pattern:

Attendance : 15 marks
Continuous Assessment : 30 marks
Internal Test (Immediately before the second series test) : 30 marks

End Semester Examination Pattern: The following guidelines should be followed regarding award of marks

(a) Preliminary work : 15 Marks
(b) Implementing the work/Conducting the experiment : 10 Marks
(c) Performance, result and inference (usage of equipments and trouble shooting) :25Marks
(d) Viva voce : 20 marks
(e) Record : 5 Marks

General instructions: Practical examination to be conducted immediately after the second series test covering entire syllabus given below. Evaluation is a serious process that is to be conducted under the equal responsibility of both the internal and external examiners. The number of candidates evaluated per day should not exceed 20. Students shall be allowed for the University examination only on submitting the duly certified record. The external examiner shall endorse the record.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Develop a program in Java to display the details of bank account using Class.
2. Develop a program in Java to implement Stack operations using Class.
3. Construct a Java class to store some employee details and provide methods to set and get values.

Course Outcome 2 (CO2)

1. Develop a Java program to implement functions to display an input integer, string and float values using the concept of method overloading

2. The Fibonacci sequence is defined by the following rule. The first 2 values in the sequence are 1, 1. Every subsequent value is the sum of the 2 values preceding it. Develop a Java program that uses both recursive and no recursive functions to print the nth value of the Fibonacci sequence

Course Outcome 3(CO3):

1. Develop a Java program to read and print students data using inheritance Class person: name, age, gender
   Class student inherits from person: mark1, mark2, mark3, total marks, grade.

2. Develop a java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea( ) that prints the area of the given shape.

3. Suppose your institution wishes to maintain a database of its employees. The database is divided into a number of classes whose (Assistant Professor, Associate Professor, Professor etc). Construct a Super Class Person to store personal information. Derive all the above classes from the Class. Develop a Java program to specify all the classes and define functions to create the database and retrieve individual information as and when required. Support at least 5 employees on each category.

Course Outcome 4 (CO4):

1. DEvelop a Java application that executes two threads. One thread displays —"Hello" in every 1000 milliseconds and other displays — "World" in every 3000 milliseconds. Create the threads by extending the Thread class

2. Construct a Stack Class with proper exception handling mechanisms. While doing a Pop operation, if the stack is empty then display an error message. While doing a Push operation, if the stack is full then display corresponding error message.

3. Develop a java program that implements a multi-thread application that has three threads. First thread generates random integer for every 1 second and if the value is even, second thread computes the square of the number and prints and if the value is odd, the third thread will print the value of cube of the number.

Course Outcome 5 (CO5):

1. Develop a java program that simulates a traffic light using AWT. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "stop" or "ready" or "go" should appear above the buttons in a selected colour. Initially there is no message shown.

2. Develop a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box

3. Develop a Java program that works as a simple calculator using SWING. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divide by zero.

LIST OF EXPERIMENTS

Cycle	Name of Experiment
I	Program to implement Simple Classes for understanding objects, member functions and Constructors. write Classes having: (i) Methods without arguments (ii) Methods with argument (iii) Constructors (iv) Methods with default arguments
II	Programs to implement (i) Method overloading (ii) Constructor overloading (iii) Static functions (iv) Inner class (v) Nested classes
III	Programs to implement Inheritance
IV	Programs to implement (i) Multi threading (ii) Exception handling (iii) Thread synchronization
V	Program to implement Graphical user Interface using: (i) AWT (i) SWING

Reference Books

1. Herbert Schildt, Java: The Complete Reference, 8/e, Tata McGraw Hill, 2011.
2. Flanagan D., Java in A Nutshell, 5/e, O'Reilly, 2005.
3. Sierra K., Head First Java, 2/e, O'Reilly, 2005.
4. Balagurusamy E., Programming JAVA a Primer, 5/e, McGraw Hill, 2014.

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITL204	DATABASE MANAGEMENT SYSTEMS LAB	PCC	0	0	3	2

Preamble:

Database Management Systems Lab course is intended to provide students a hands on experience in database management concepts. It also provides astrong formal foundation in database concepts, technology and practice to the students. It gives an exposure to design and develop applications.

Prerequisite:

CODE	COURSE NAME	DESCRIPTION	SEM
ITT206	Database Management Systems	Gives concepts of database management systems and exposure to database programming, modelling and design.	4

Course Outcomes: After completion of the course, the student will be able to

CO No.	DESCRIPTION	Blooms' Taxonomy
CO1	Construct database using DDL, DCL and basic DML commands in SQL.	Apply
CO2	Build nested and join queries.	Apply
CO3	Applyprocedural SQL concepts like view, exception handling, stored procedure, function, trigger, cursor in various database applications.	Apply
CO4	Design and develop database applications.	Create

Mapping of Course Outcomes with Program Outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
CO1	2	1	2	-	1	-	-	-	-	-	-	1
CO2	2	2	1	1	1	-	-	-	-	-	-	-

CO3	3	2	2	1	-	-	-	-	-	-	-	-
CO4	2	3	3	2	-	-	-	-	-	-	-	-

3/2/1: High/Medium/Low

Assessment Pattern

Mark distribution

Total Marks	**CIE**	**ESE**	**ESE Duration**
150	75	75	2.5 hours

Continuous Internal Evaluation Pattern:

Attendance :15 marks
Continuous Assessment :30 marks
Internal Test (Immediately before the second series test) :30 marks

End Semester Examination Pattern: The following guidelines should be followed regarding award of marks

1.
   a. Database Design : 10 Marks
   b. Implementation of Project : 15 Marks
2. Performance, result and inference
   (usage of application tool and trouble shooting) : 25 Marks
3. Viva voce : 20 Marks
4. Record : 5 Marks

General instructions: Practical examination to be conducted immediately after the second series test covering entire syllabus given below. Evaluation is a serious process that is to be conducted under the equal responsibility of both the internal and external examiners. The number of candidates evaluated per day should not exceed 20. Students shall be allowed for the University examination only on submitting the duly certified record. The external examiner shall endorse the record.

LIST OF EXPERIMENTS (Experiments No.10 & 11 are not mandatory)

1. Familiarization of Data Definition Language (DDL)and Data Control Language (DCL) commands.
2. Familiarization ofData Manipulation Language (DML) commands (INSERT, SELECT, DELETE and UPDATE).
3. Implementation of various Aggregate functions and Groupingin SQL.
4. Implementation of Nested Queries.
5. Implementation of Join Queries.
6. Creation of Views.
7. Creation of Stored Procedures and Functions.
8. Exception Handling in SQL.
9. Creation of Triggers and Cursors.
10. Familiarization of Transaction Control Language(TCL) Commands.
11. Familiarization of NoSQL databaseusing MongoDB.
12. Develop an application to demonstrate database connectivity.

CLASS PROJECT (One project per group of at most four members)

Applications like Library Management System, Hospital Management System, Student Management Systems, Reservation Systems etc. can be considered as project topics.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Create a table project and for each project retrieve the project number, project name and the number of employees who work on that project.
2. Retrieve the social security number of all employees who work on project number 1,2 or 3

Course Outcome 2 (CO2)

3. Create a table employee with employee number, name, SSN, salary and department number and display the minimum salary of employee whose salary is greater than salary of all employees in department 5.

Course Outcome 3(CO3):

4. Createa SQL procedure application for exception using continue and exit handler

5. Create a table and perform cursor operations and trigger

Course Outcome 4 (CO4):

6. Develop a data driven GUI application in any domain (bank, library, hospital etc.)
    a. Implementation of student management system
    b. Implementation of any reservation system

Text Books

1. Abraham Silberschatz, Henry F. Korth and S. Sudarshan, Database System Concepts, McGraw-Hill Education (Asia), Fifth Edition, 2006.
2. AtulKahate, Introduction to Database Management Systems, Pearson
3. C. J. Date, A. Kannan and S. Swamynathan, An Introduction to Database Systems, Pearson Education, Eighth Edition, 2009.
4. Patrick O'Neil and Elizabeth O'Neil, Database Principles, Programming and Performance, Harcourt Asia Pte. Ltd., First Edition, 2001.
5. Peter Rob and Carlos Coronel, Database Systems Design, Implementation and Management, Thomson Learning-Course Technology, Seventh Edition, 2007.
6. RamezElmasri, Shamkant B. Navathe, Fundamentals of Database Systems (7th Edition), Pearson Education Ltd.
7. Shio Kumar Singh, Database Systems Concepts, Designs and Application, Pearson Education, Second Edition, 2011.

SEMESTER -4

MINOR

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT282	DATABASE MANAGEMENT	VAC	3	1	0	4

Preamble: This course aims at facilitating the student to understand the various functionalities of DBMS software and perform many operations related to creating, manipulating and maintaining databases for Real-world applications and student to understand the various designing concepts, storage methods, querying and managing databases.

Prerequisite: Nil

Course Outcomes: After the completion of the course the student will be able to

CO_NO	Course Outcome(CO)	Bloom's Category
CO 1	Impart the basic understanding of the theory and applications of database management systems.	Level 2 : Understand
CO 2	Give basic level understanding of internals of database systems.	Level 2 : Understand
CO 3	Construct simple and moderately advanced database queries using Structured Query Language (SQL)	Level 3: Apply
CO 4	Understand and successfully apply logical database design principles and database normalization.	Level 3: Apply
CO 5	Give understanding of organization of Physical Data in DBMS and expose to some of the recent trends in databases	Level 2 : Understand

Mapping of course outcomes with program outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	2	1	-	-	-	-	-	-	-	-	-
CO 2	2	1	2	1	-	-	-	-	-	-	-	-
CO 3	3	2	3	2	1	-	-	-	-	-	-	1
CO 4	3	2	3	2	1	-	-	-	-	-	-	1
CO 5	1	1	3	3	1	-	-	-	-	-	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	20
Understand	20	20	40
Apply	20	20	40

Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Compare structured data and unstructured data?
2. Explain the advantages of DBMS?
3. Relate Entity, Entity Set, and Entity Type?

Course Outcome 2 (CO2)

1. Explain about Integrity Constraints
2. Describe views in DBMS?
3. Explain the purpose of DML commands in SQL

Course Outcome 3(CO3):

1. Develop SQL queries.
2. Experiment with the use of group by and having clause in sql

3. Build SQL query to find the name of the maximum salaried employee in each department

Course Outcome 4 (CO4):

1. Let R= (A, B, C, D, E, F) be a relation scheme with the following dependencies:

   C->F, E->A, EC->D, A->B. Identify the key for R?

2. Experiment with 3NF with example

3. Make use of lossless and dependency preserving decompositions?

Course Outcome 5 (CO5):

1. Explain the use of Query Optimization
2. Compare Non-clustered and clustered index
3. Explain the concept of ACID properties in DBMS
4. Explain Semantic Web, RDF, GIS

Model Question paper

Course Code: ITT282

Course Code: DATABASE MANAGEMENT

Max Marks:100 Duration: 3hr

PART A

Answer all questions, each carries 3 marks

1. List any three categories of database users, highlighting any one important characteristic of each category.

2. In a relationship of degree 2, how can we decide if an attribute of the relationship can be moved to one of the entity sets?

3. Distinguish between total and partial participation constraints with the help of real examples.

4. Illustrate DELETE and UPDATE clauses using typical examples.

5. Given a relation R(A,B,C,D,E,F) with functional dependencies A→B, B→D, D→EF, F→A, compute {D}+ and {EF}+.

6. What are fully functional dependencies and partial functional dependencies? Give an example to distinguish between these?

7. Define the following:

(a) physical record

(b) logical record

(c) blocking factor

8. How is clustering index different from primary index?

9. What are the desirable properties of transactions? Explain.

10. What is the significance of check-pointing. (10*3=30)

PART B

Answer all questions, each carries 14 marks

11. a) Design an ER diagram to represent the following scenario: A company has many employees working on a project. An employee can be part of one or more projects. Each employee works on a project for certain amount of time. Assume suitable attributes for entities and relations. Mark the primary key(s) and the cardinality ratio of the relations. (9)

b) What are logical data independence and physical data independence? What is the difference between them? Which of these harder to realize? Why? (5)

OR

12. a) With the help of neat diagram, explain three schema architecture of DBMS. (9)

b) How is weak entity type different from a strong entity type? Give an example. (5)

13. a) With the help of an example, compare DML and DDL. (6)

b) Consider the following relational schema:

Suppliers(sid:integer, sname:string, city:string, street:string)

Parts(pid:integer, pname:string, color:string)

Catalog(sid:integer, pid:integer, cost:real)

Write SQL query to find the names of all suppliers who have not supplied only blue parts.(8)

OR

14. a)Explain the aggregate functions in SQL? (6)

b) Consider the following relational schema:

employee(empId, empName, empDept)

customer(custId, custName, salesRepId, rating)

salesRepId is a foreign key referring to empId of the employee relation. Assume that each employee makes a sale to at least one customer. Write SQL query to find the names of all the employees with all their customers having a 'GOOD' rating. (8)

15. a) Explain three uses of attribute closure algorithm (5)

    b) Given a relation R(A,B,C,D,E,F,G, H) with keys BD and C and functional dependencies D→G, E→F and H→C, decompose the R into the highest normal form possible. (9)

OR

16. a) What are Armstrong's axioms (5)

    b) Given a relation R(A1,A2,A3,A4,A5) with functional dependencies A1→A2A4 and A4→A5, check if the decomposition R1(A1,A2,A3), R2(A1,A4), R3(A2,A4,A5) is lossless. (9)

17. a) Illustrate structure of B-Tree and B+-Tree and explain how they are different. (5)

    b) Consider an EMPLOYEE file with 10000 records where each record is of size 80 bytes. The file is sorted on employee number (15 bytes long), which is the primary key. Assuming un-spanned organization, block size of 512 bytes and block pointer size of 5 bytes, compute the number of block accesses needed for retrieving an employee record based on employee number if (i) No index is used (ii)Multi-level primary index is used. (9)

OR

18. a) Distinguish between dense index and sparse index (5)

    b) Explain heuristics-based query optimization. (9)

19. a) Check if the following schedules are conflict-serializable using precedence graph. If so, give the equivalent serial schedule(s). r3(X), r2(X), w3(X), r1(X), w1(X). (7)

    b) Explain the concept behind the following:

    (i) Log base recovery.

    (ii) Deferred database modification. (7)

OR

20. a) Why Concurrency Control Is Needed? What are the different types of problems we may encounter when two transactions run concurrently? Illustrate each problem with suitable examples. (7)

    b)Explain the characteristics of data in GIS. (7)

Syllabus

Module 1 (7 Hours)
Introduction: Data: structured, semi-structured and unstructured data, Concept & Overview of DBMS, Data Models, Database Languages, Database Administrator, Database Users, Three Schema architecture of DBMS. Database architectures and classification. Entity-Relationship Model: Basic concepts, Design Issues, Mapping Constraints, Keys, Entity-Relationship Diagram, Weak Entity Sets.
Module 2 (8 Hours)
Relational Model: Structure of relational Databases, Integrity Constraints, synthesizing ER diagram to relational schema .**Database Languages:** Concept of DDL and DML relational algebra. **Structured Query Language (SQL**): Basic SQL Structure, Set operations, Aggregate Functions, nested sub-queries, Views, assertions and triggers.
Module 3 (10 Hours)
Relational Database Design: Different anomalies in designing a database, normalization, functional dependency (FD), Armstrong's Axioms, closures, Equivalence of FDs, minimal Cover. Normalization using functional dependencies, INF, 2NF, 3NF and BCNF, lossless and dependency preserving decompositions.
Module 4 (10 Hours)
Physical Data Organization: index structures, primary, secondary and clustering indices, Single level and Multi-level indexing, B+- Trees .Query Optimization: heuristics-based query optimization.
Module 5 (10 Hours)
Transaction Processing Concepts: overview of concurrency control and recovery acid properties, serial and concurrent schedules, conflict serializability, Two-phase locking, failure classification, storage structure, stable storage, log based recovery, deferred database modification, check-pointing, **Recent topics :** Semantic Web and RDF, GIS, biological databases .

Text Books

1. Elmasri R. and S. Navathe, Database Systems: Models,Languages,Design and Application Programming, 6e,2013.
2. Sliberschatz A., H. F. Korth and S. Sudarshan, Database System Concepts, 6/e, McGraw Hill, 2011.
3. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, McGrawHill Education, 3rd Edition, 2003.

Reference Books

1. Powers S., *Practical RDF*, O'Reilly Media, 2003.

2. Plunkett T., B. Macdonald, *et al.*, *Oracle Big Data Hand Book*, Oracle Press, 2013.
3. Peter Rob and Carlos Coronel, Database System- Design, Implementation and Management (7/e), Cengage Learning, 2007.

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction**	**7 Hours**
1.1	Data: structured, semi-structured and unstructured data, Concept & Overview of DBMS	2
1.2	Data Models, Database Languages, Database Administrator, Database Users, Three Schema architecture of DBMS, Database architectures and classification.	2
1.3	Entity-Relationship Model: Basic concepts, Design Issues, Mapping Constraints, Keys, Entity- Relationship Diagram, Weak Entity Sets.	3
2	**Relational Model**	**8 Hours**
2.1	Structure of relational Databases, Integrity Constraints, synthesizing ER diagram to relational schema	3
2.2	Database Languages, Concept of DDL and DML relational algebra.	2
2.3	Basic SQL Structure, Set operations, Aggregate Functions, nested sub-queries, Views, assertions and triggers.	3
3	**Relational Database Design**	**10 Hours**
3.1	Different anomalies in designing a database, normalization	3
3.2	Functional dependency (FD), Armstrong's Axioms, closures, Equivalence of FDs, minimal Cover.	3
3.3	Normalization using functional dependencies, INF, 2NF, 3NF and BCNF, lossless and dependency preserving decompositions.	4
4	**Physical Data Organization**	**10 Hours**
4.1	index structures, primary, secondary and clustering indices	3
4.2	Single level and Multi-level indexing, B+- Trees	3
4.3	Query Optimization: heuristics-based query optimization.	4
5	**Transaction Processing Concepts**	**10 Hours**
5.1	overview of concurrency control and recovery, acid properties, serial and concurrent schedules, conflict serializability	3
5.2	Two-phase locking, failure classification, storage structure, stable storage, log based recovery, deferred database modification, check-pointing	4
5.3	Semantic Web and RDF, GIS, biological databases.	3

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT284	COMPUTER NETWORKS	VAC	3	1	0	4

Preamble: The syllabus is prepared with a view to equip the Engineering Graduates to learn basic concepts in computer networking, and to design, inspect and evaluate network models and protocols for real world applications.

Prerequisite: Nil

Course Objectives

- To introduce the basic terminology and concepts used in computer networking
- To understand data link layer services and protocols
- To learn and apply the process of routing and IP addressing in Internet

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Examine different network design models and protocols	Level 1: Remember
CO 2	Inspect data link layer issues and protocols	Level 2: Understand
CO 3	Apply the process of routing and IP addressing in Internet	Level 3: Apply
CO 4	Understand transport layer services and congestion control mechanisms	Level 2: Understand
CO 5	Demonstrate the working of various application layer protocols such as HTTP, SMTP, POP3, FTP and DNS. Explain various Internet control protocols used to manage and monitor network traffic.	Level 2: Understand

Mapping of Course Outcomes with Program Outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	2	-	-	-	-	-	-	-	-	2
CO 2	3	2	-	2	-	-	-	-	-	-	-	-
CO 3	3	3	3	2	1	-	-	-	-	-	-	3
CO 4	2	-	2	2	-	-	-	-	-	-	-	-
CO 5	2	-	2	3	-	-	-	-	-	-	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category Levels	Continuous Assessment Tests		End Semester Examination
	1	2	
BL 1: Remember	10	10	20
BL 2: Understand	30	30	60
BL 3: Apply	10	10	20
BL 4: Analyse			
BL 5: Evaluate			
BL 6: Create			

Mark distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be *two* parts; **Part A** and **Part B**. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer *all* questions. Part B contains 2 questions from each module of which student should answer *any one*. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. List the various layers of the OSI reference model.
2. What are the different types of network topologies?
3. What are the various devices used in different layers of the TCP/IP reference model.
4. Define a Protocol Data Unit (PDU).

Course Outcome 2 (CO 2):

1. Compare and contrast the functionalities of hubs, bridges and switches.
2. Explain the main features of Fast Ethernet?
3. Explain the different fields in IEEE 802.3 frame format?

4. Explain the techniques for detecting burst errors in data transmission.

Course Outcome 3 (CO 3):

1. Experiment with the working of Distance Vector Routing algorithm.
2. A block is assigned an IP address 201.99.88.119/22. Identify the IP address of the first and last host of this block.
3. What is super-netting? What is its application in classless addressing?
4. What is the relevance of Token Bucket algorithm in computer networks?

Course Outcome 4 (CO 4):

1. Explain the appropriateness of using a pseudo-header in TCP for computing checksum.
2. Illustrate the steps involved in TCP connection establishment and release.
3. Explain with the help of an example, the working of Remote Procedure Calls.
4. Describe the various congestion control mechanisms in transport layer.

Course Outcome 5 (CO 5):

1. Explain how a Domain Name System (DNS) works.
2. Compare link state routing algorithms with distance vector routing protocols
3. Explain the suitability of various error correcting codes to deal with single-bit and burst errors in data transmission.
4. Describe the major design issues in network layer.
5. Describe the working of SMTP, IMAP and POP3 mail transfer protocols.

Model Question Paper

Part A

Answer all questions. Each question carries 3 marks. (10 * 3 = 30 Marks)

1. List the main characteristics of different types of Computer Networks.
2. Define Maximum Transmission Unit (MTU) of a protocol data unit.
3. Briefly explain the various types of CSMA protocols.
4. A message 11001001 has to be transmitted using the CRC polynomial $x^3 +1$ to protect it from errors. Compute the message that should actually be transmitted.
5. How does link state routing build and distribute the link state packets?
6. One of the IP addresses of a block of address is 201.99.88.119/22. Find the range of the assignable IP address.

7. What is traffic shaping? Compare traffic shaping with traffic policing.
8. Why is Token Bucket algorithm relevant in networks?
9. How can ARP and RARP be used to resolve addresses in computer networks?
10. What happens when an FTP control connection breaks while data transfer is in progress?

Part B

Answer all questions. Each question carries 14 marks. (5 * 14 = 70 Marks)

11. List and explain the main features of all the seven layers of the TCP/IP reference model.

OR

12. Explain the major design issues for the network layer.
13. Why do Ethernet frames require a minimum frame size? Discuss how Gigabit Ethernet solves this problem.

OR

14. How do burst errors occur? Explain how Hamming codes can be effectively used to deal with burst errors.
15. Compute the IP address of the 5^{th} host in the 6^{th} subnet of a network whose network address is given by 192.168.0.1 and subnet mask is given by 255.255.255.240.

OR

16. What is count-to-infinity problem? Discuss any two methods to practically solve this problem.
17. Illustrate with suitable examples, the operation of Go-Back-N and Selective-Repeat sliding window protocols.

OR

18. Describe the TCP segment header format. Clearly indicate the significance of each flag.
19. Differentiate between persistent and non-persistent HTTP connections. Discuss the message formats of HTTP request and response.

OR

20. Describe the working of SMTP, IMAP and POP3 protocols in a simple mail transfer scenario.

Syllabus

Module 1 (7 Hours)
Computer Networks - Types of Networks, Reference models - OSI and TCP/IP, Internet - The network edge, The network core, Network access, Delay and loss, Protocol layers and services - Design issues for the layers - Interface and Services
Module 2 (9 Hours)
Data Link layer design Issues - Flow Control and ARQ techniques, Services - Error detection and correction, Protocols - HDLC, MAC, Multiple access protocols, MAC Sub layer - IEEE 802 for LANs and MANs, IEEE 802.3, 802.4, 802.5, Devices - Hubs, Bridges and Switches, VLAN, High-speed LANs - Gigabit Ethernet.
Module 3 (10 Hours)
Network layer - services, IPv4 - IP Addressing - Classless and Classfull Addressing. Sub-netting and super-netting, Routing in Internet - Shortest path routing, Flooding, Distance Vector Routing, Link State Routing, RIP and OSPF, IPV6, Internet Multicasting, Multicast routing.
Module 4 (10 Hours)
Transport layer services and primitives, UDP - Segment Structure, Remote Procedure Call, TCP - Segment Header, Connection establishment and Release, Transmission Policy, Congestion Control - General principles, Quality-of-Service requirements - Traffic shaping.
Module 5 (9 Hours)
Internet Control and Management Protocols - ICMP, SNMP, ARP and RARP, Application Layer - HTTP - Overview, Persistent and non-persistent connections, Message formats, Cookies, FTP, Electronic Mail - SMTP, POP3 and IMAP, DNS - services and caching policies.

Text Books

1. F. Kurose and K. W. Ross, Computer Networking: A Top-Down Approach Featuring Internet, 6/e, Pearson Education, 2012.
2. A. S. Tanenbaum and D. J. Wetherall, Computer Networks, 5/e, Pearson, 2013.
3. L. L. Peterson and B. S. Davie, Computer Networks, A systems approach, 5/e, Morgan Kaufmann, 2011.

Reference Books

1. William Stallings, Computer Networking with Internet Protocols, Prentice-Hall, 2004.
2. Behrouz A. Forouzan, TCP/IP Protocol Suite, 4/e, Mc Graw Hill

3. Behrouz A. Forouzan, Data Communications and Networking, 4/e, Tata McGraw Hill.
4. Fred Halsall, Computer Networking and the Internet, 5/e.

Course Contents and Lecture Schedule

Sl. No.	Topic	No. of Lectures
1	**Computer Networks – Fundamentals**	**7 Hours**
1.1	Types of Networks, Reference models - OSI and TCP/IP.	2
1.2	Internet - The network edge, The network core, Network access, Delay and loss.	2
1.3	Protocol layers and services - Design issues for the layers - Interface and Services.	3
2	**Data Link layer**	**9 Hours**
2.1	Design Issues - Flow Control and ARQ techniques, Services - Error detection and correction.	3
2.2	Protocols - HDLC, MAC, Multiple access protocols, MAC Sub layer - IEEE 802 for LANs and MANs.	3
2.3	IEEE 802.3, 802.4, 802.5, Devices - Hubs, Bridges and Switches, VLAN, High-speed LANs - Gigabit Ethernet.	3
3	**Network layer**	**10 Hours**
3.1	Services, IPv4 - IP Addressing.	3
3.2	Classless and Classfull Addressing. Sub-netting and super-netting	3
3.3	Routing in Internet - Shortest path routing, Flooding, Distance Vector Routing, Link State Routing, RIP and OSPF, IPV6, Internet Multicasting, Multicast routing.	4
4	**Transport layer**	**10 Hours**
4.1	Services and primitives, UDP - Segment Structure, Remote Procedure Call.	3
4.2	TCP - Segment Header, Connection establishment and Release, Transmission Policy.	3
4.3	Congestion Control - General principles, Quality-of-Service requirements - Traffic shaping.	4
5	**Application layer**	**9 Hours**
5.1	Internet Control and Management Protocols - ICMP, SNMP, ARP and RARP.	3
5.2	Application Layer - HTTP - Overview, Persistent and non-persistent connections, Message formats, Cookies.	2
5.3	FTP, Electronic Mail - SMTP, POP3 and IMAP, DNS - services and caching policies.	4

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT286	SOFTWARE PROJECT MANAGEMENT TECHNIQUES	VAC	3	1	0	4

Preamble: The syllabus is prepared with the view of preparing the Engineering Graduates capable of understanding essential concept of software project management and software development process.

Prerequisite: Basics of programming, software engineering.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category
CO1	**List** the activities in Software Project Management.	Level 1: Remember
CO2	**Summarize** different Software Process Models	Level 2: Understand
CO3	**Explain** methods for software cost estimation	Level 2: Understand
CO4	**Analyze** Project Scheduling and risk management methods.	Level 3: Apply
CO5	**Illustrate** the methods to manage and control projects and people in an organization.	Level 2: Understand

Mapping of course outcomes with program outcomes

COs	PROGRAMME OUTCOMES (PO)											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	1	-	1	-	-	2	-	3	1	3	1
CO2	-	-	-	-	-	-	-	-	2	2	3	1
CO3	2	1	-	1	2	-	-	-	3	3	3	1
CO4	1	3	-	-	1	-	-	-	3	3	3	1
CO5	-	1	-	1	1	-	3	-	2	2	3	1

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	20	20
Apply	20	20	70
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Test (2 numbers) : 25 marks

Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. List the characteristics which make software projects different from other project.
2. List out the activities within stepwise planning
3. What are the various activities covered by software project management.
4. Define portfolio management of software projects. Discuss with examples

Course Outcome 2 (CO2):

1. Illustrate SRUM in details
2. Explain the phases of software development
3. Describe the rapid software development method
4. Explain Agile model in detail.

Course Outcome 3 (CO3):

1. Write short note on reliability metrics.
2. Write the steps in cost-benefit analysis comprises
3. Classify the project sizing metrics.
4. Describe COCOMO model with appropriate data

Course Outcome 4 (CO4):

1. Experiment with network planning models.
2. Experiment with the different steps in project scheduling.
3. Make use of resource allocation and cost scheduling methods in software projects.

Course Outcome 5 (CO5):

1. Explain the methods to visualize the progress of the project.
2. Summarize the setting of checkpoints.
3. Illustrate the salient features of fixed price and time and material contract models
4. Explain the procedure of change control
5. Compare intrinsic and extrinsic motivation.
6. Explain the Oldham-Hackman job characteristic model
7. Explain in detail about the team structures

Model Question Paper

PART A **(10*3=30)**

(Each Question carries 3 Marks)

1. Identify the characteristics which make software projects different from other project.
2. List out the activities within stepwise planning
3. Examine the steps in cost-benefit analysis comprises
4. Classify the project sizing metrics.
5. Illustrate network planning models with example.
6. Define critical path.
7. Identify the methods to visualize the progress of the project.
8. Summarize the setting of checkpoints.
9. Differentiate intrinsic and extrinsic motivation.
10. List some obstacles for good group decision making

PART B **(5*14=70)**

11. Explain the various activities covered by software project management.

OR

12. Explain portfolio management of software projects. Discuss with examples.
13. Illustrate SRUM in details

OR

14. Describe COCOMO model with appropriate data.
15. Discuss the steps in project scheduling.

OR

16. Explain resource allocation and cost scheduling in software projects.
17. Illustrate the salient features of fixed price and time and material contract models

OR

18. Elaborate on the procedure of change control
19. Discuss in detail about the team structures

OR

20. Explain the Oldham-Hackman job characteristic model.

Syllabus

Module 1: Project Planning (8 hrs)
Introduction to Software Project Management, Management tasks
Module 2: Project Cost Estimation (10 hrs)
Software process and Process Models, Cost Estimation
Module 3: Project Scheduling and Risk Management (11 hrs)
Project schedules, Critical Path Analysis, Risk identification
Module 4: Project Management And Control (9 hrs)
Framework for Management and control, Analysis and Project tracking
Module 5: Project Staffing (7 hrs)
Managing people, methods of staff selection, Team structures

Text Books

T1. Bob Hughes, Mike Cotterell and Rajib Mall: Software Project Management – Sixth Edition, Tata McGraw Hill, New Delhi, 2017

Reference Books

R1. Roger S. Pressman, Software Engineering: A practitioner's approach, 8th Edition (Indian Edition), McGraw Hill. 2019

R2. Harold Kerzner,Program Management-A System Approach Planning Scheduling And Controlling, 12th Edition, Wiley, 2017.

R3. Sunitha E.V, Sarath K.S, Software Project Management, Jyothis Publishers 2019.

R4. Jack Marchewka, Information Technology Project Management 5th edition. John Wiley & Sons (2012). ISBN: 978-1-118-91101-3. 2016.

Course Contents and Lecture Schedule

Sl. No	Module 1: Project Planning	8hrs
1.1	Introduction to Software Project Management – importance, Activities, Methodologies – types of Software Projects – Setting objectives.	2
1.2	Management Principles – Management Control – Project portfolio Management	2
1.3	Cost-benefit evaluation technology – Risk evaluation	2
1.4	Strategic program Management – Stepwise Project Planning.	2
	Module 2 Project Cost Estimation	**10 hrs**
2.1	Software process and Process Models – Rapid Application development – Agile methods	3
2.2	Extreme Programming – SCRUM – Managing interactive processes.	2
2.3	Basics of cost estimation – Effort and Cost estimation techniques – LOC, FP, COSMIC Full function points	3
2.4	COCOMO models - A Parametric Productivity Model.	2
	Module 3: Project Scheduling and Risk Management	**11 hrs**
3.1	Objectives of Activity planning – Project schedules – Activities	1
3.2	Sequencing and scheduling –Network Planning models – Forward Pass & Backward Pass techniques – Critical path (CRM) method.	3
3.3	Risk identification – Assessment – Monitoring – PERT technique	2
3.4	Monte Carlo simulation –Resource Allocation	2
3.5	Creation of critical patterns – Cost schedules.	3

	Module 4: Project Management And Control	**9hrs**
4.1	Framework for Management and control – Collection of data.	2
4.2	Project termination – Visualizing progress – Cost monitoring	2
4.3	Earned Value Analysis- Project tracking	2
4.4	Change control - Software Configuration Management	2
4.5	Managing contracts	1
	Module 5: Project Staffing	**7 hrs**
5.1	Managing people – Organizational behavior	1
5.2	Methods of staff selection – Motivation – The Oldham-Hackman job characteristic model	3
5.3	Ethical and Programmed concerns – Working in teams – Decision making – Team structures – Virtual teams – Communications genres – Communication plans.	3

SEMESTER -4
HONOURS

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT292	**MATHEMATICAL FOUNDATION FOR NETWORKING**	**VAC**	3	1	0	4

Preamble: The course is intended to provide the concepts of random variable, random processes and probability distribution. It also covers the basics of queuing theory and linear programming techniques.

Prerequisite: Background in calculus and linear algebra.

Course Outcomes: After the completion of the course the student will be able to

	Course outcomes	Bloom's Category
CO 1	Express linear programming and apply simplex method linear programming problem	Apply
CO 2	Apply sensitivity analysis on LPP and solve transportation problems	Apply
CO 3	Summarizethe characteristics of random processes and demonstrate the applications of Poisson processes.	Understand
CO 4	Compare and Contrast the various queuing models	Understand
CO 5	Apply the queuing theory on different applications	Apply

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3	3	2	-	-	1	-	1	-	1	3
CO 2	3	3	3	2	-	-	1	-	1	-	1	2
CO 3	3	3	3	2	-	-	1	-	1	-	1	2
CO 4	3	3	3	2	-	-	1	-	1	-	1	2
CO 5	3	3	3	2	-	-	1	-	1	-	1	3

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	20	20	20
Understand	20	20	70
Apply	10	10	10
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

A progressive university has decided to keep its library open round the clock and gathered that the following numbers of attendants are required to re-shelve the books:

Time of day (hours)	Minimum number of attendants required
0–4	4
4–8	7
8–12	8
12–16	9
16–20	14
20–24	3

If each attendant works eight consecutive hours per day, formulate the problem of finding the minimum number of attendants necessary to satisfy the requirements above as a LP problem.

Course Outcome 2 (CO2)

Find the solution of the following problem using Karmarkar's method:

Minimize $f = 2x_1 + x_2 - x_3$

subject to:

$x_2 - x_3 = 0$

$x_1 + x_2 + x_3 = 1$

$x_i \geq 0, i = 1, 2, 3$

Course Outcome 3(CO3):

Cars arrive at a gas station according to a Poisson process at an average rate of 12 cars per hour. The station has only one attendant. If the attendant decides to take a 2-minute coffee break when there are no cars at the station, what is the probability that one or more cars will be waiting when he comes back from the break, given that any car that arrives when he is on coffee break waits for him to get back?

Course Outcome 4 (CO4):

Consider an airport runway for arrivals only. Arriving aircraft join a single queue for the runway. Where, the service time is exponentially distributed with a rate $\mu = 27$ arrivals / hour . And the Poisson arrivals with a rate $\lambda = 20$ arrivals / hour.

i. What will be the quantities of the queuing system?

ii. Suppose we are in holidays and the arrival rate increases $\lambda = 25$ arrivals / hour How will the quantities of the queuing system change?

iii. Now suppose we have a bad weather and the service rate decreases $\mu = 22$ arrivals / hour ‰ How will the quantities of the queuing system change?

Course Outcome 5 (CO5):

Consider a steady state open network with three exponential nodes with parameters ($\mu 1$, $\mu 2$, $\mu 3$) and Poisson arrivals to node 1. Customers follow one of two routes through the network: node 1 to node 2 (with probability p) and node 2 to node 3 (with probability q=1-p). Write down the arrival rates λi at node i (i=1, 2, 3). Use Little's theorem and Jackson's theorem to obtain the mean waiting time spent by a customer in the network and show that if $\mu 2 = \mu 3$, this is least when p = q =1/2.

Model Question paper

Course Code: ITT292

Course Name: MATHEMATICAL FOUNDATION FOR NETWORKING

Max.Marks:100 **Duration: 3 Hours**

PART A

Answer all Questions. Each question carries 3 Marks

1. How do you solve a maximization problem as a minimization problem?

2. How many basic solutions can an LP problem have? Why?

3. State the following LP problem in standard form:

$$\text{Maximize } f = -2x_1 - x_2 + 5x_3$$

subject to

$$x_1 - 2x_2 + x_3 \leq 8$$
$$3x_1 - 2x_2 \geq -18$$
$$2x_1 + x_2 - 2x_3 \leq -4$$

4. Write the dual of the following linear programming problem:

$$\text{Maximize } f = 50x_1 + 100x_2$$

subject to

$$2x_1 + x_2 \leq 1250$$
$$2x_1 + 5x_2 \leq 1000$$
$$2x_1 + 3x_2 \leq 900$$
$$x_2 \leq 150$$

where

$$x_1 \geq 0 \text{ and } x_2 \geq 0$$

5. University buses arrive at the Students' Centre to take students to their classes according to a Poisson process with an average rate of 5 buses per hour. Chris just missed the last bus. What is the probability that he waits more than 20 minutes before boarding a bus?

6. Calculate the autocorrelation function of the periodic function $X(t) = A\sin(\omega t + \varphi)$, where the period $T = 2\varphi/\omega$, and A, φ, and ω are constants.

7. Prove that the exponential distribution has both the lack of memory and the minimum property.

8. A monitor on a disk server showed that the average time to satisfy an I/O request was 100 milliseconds. The I/O rate was about 100 requests per second. What was the mean number of requests at the disk server?

9. Draw the state transition rate diagram of an M/M/C queueing model.

10. What do you mean by balking and reneging?

PART B

Answer any one Question from each module. Each question carries 14 Marks

11. a. Find the solution of the following LP problem graphically: **(7 Marks)**

$$\text{Minimize } f = 3x_1 + 2x_2$$

subject to

$$8x_1 + x_2 \geq 8$$
$$2x_1 + x_2 \geq 6$$
$$x_1 + 3x_2 \geq 6$$
$$x_1 + 6x_2 \geq 8$$
$$x_1 \geq 0,\ x_2 \geq 0$$

b. Prove that the feasible region of a linear programming problem is convex. **(7 Marks)**

OR

12. A manufacturer produces three machine parts, A, B, and C. The raw material costs of parts A, B, and C are \$5, \$10, and \$15 per unit, and the corresponding prices of the finished parts are \$50, \$75, and \$100 per unit. Part A requires turning and drilling operations, while part B needs milling and drilling operations. Part C requires turning and milling operations. The number of parts that can be produced on various machines per day and the daily costs of running the machines are given below:

Machine part	Number of parts that can be produced on		
	Turning lathes	Drilling machines	Milling machines
A	15	15	
B		20	30
C	25		10
Cost of running the machines per day	\$250	\$200	\$300

Formulate the problem of maximizing the profit. **(14Marks)**

13. A metallurgical company produces four products, A, B,C, and D, by using copper and zinc as basic materials. The material requirements and the profit per unit of each of the four products, and the maximum quantities of copper and zinc available are given below:

	Product				Maximum quantity available
	A	*B*	*C*	*D*	
Copper (lb)	4	9	7	10	6000
Zinc (lb)	2	1	3	20	4000
Profit per unit (\$)	15	25	20	60	

a. Find the number of units of the various products to be produced for maximizing the profit.
b. Find the effect of changing the profit per unit of product D to \$30.
c. Find the effect of changing the available quantities of copper and zinc to 4000 and 6000 lb, respectively.
d. If product C requires 5 lb of copper and 4 lb of zinc (instead of 7 lb of copper and 3 lb of zinc) per unit, find the change in the optimum solution.

(14Marks)

OR

14. The Childfair Company has three plants producing child push chairs that are to be shipped to four distribution centers. Plants A, B, and C produce 12, 17, and 11 shipments per month, respectively. Each distribution center needs to receive 10 shipments per month. The distance from each plant to the respective distributing centers is given below:

	Distance			
	Distribution Centres			
	1	2	3	4

Plants	A	80 KM	130 KM	40 KM	70 KM
	B	110 KM	140 KM	60 KM	100 KM
	C	60 KM	120 KM	80 KM	90 KM

The freight cost for each shipment is Rs.100 per Kilometer. How much should be shipped from each plant to each of the distribution centers to minimize the total shipping cost?

a. Formulate this problem as a transportation problem by constructing the appropriate parameter table.
b. Draw the network representation of this problem.
c. Obtain an optimal solution.

(14 Marks)

15. Alan is conducting an experiment to test the mean lifetimes of two sets of electric bulbs labelled A and B. The manufacturer claims that the mean lifetime of bulbs in set A is 200 hours, while the mean lifetime of the bulbs in set B is 400 hours. The lifetimes for both sets are exponentially distributed. Alan's experimental procedure is as follows: He started with one bulb from each set. As soon as a bulb from a given set fails (or burns out), he immediately replaces it with a new bulb from the same set and writes down the lifetime of the burnt-out bulb. Thus, at any point in time he has two bulbs on, one from each set. If at the end of the week Alan tells you that 8 bulbs have failed, determine the following:

b. The probability that exactly 5 of those 8 bulbs are from set B.
c. The probability that no bulb will fail in the first 100 hours.
d. The mean time between two consecutive bulb failures. **(14Marks)**

OR

16. Two random processes X(t) and Y(t) are defined as follows: **(14Marks)**

$$X(t) = A\cos(\omega 1t + \Theta)$$
$$Y(t) = B\sin(\omega 2t + \Phi)$$

where ω1, ω2, A, and B are constants, and Θ and Φ are statistically independent random variables, each of which is uniformly distributed between 0 and 2π.

a. Find the cross correlation function $R_{XY}(t, t + \tau)$, and show that X(t) and Y(t) are jointly wide-sense stationary.
b. If Θ = Φ, show that X(t) and Y(t) are not jointly wide-sense stationary.
c. If Θ = Φ, under what condition are X(t) and Y(t) jointly wide-sense stationary?

17. a. An airport has a single runway. Airplanes have been found to arrive at the rate of 15 per hour. It is estimated that each landing takes 3 minutes. Assuming a Poisson process for arrivals and an exponential distribution for landing times. Find the expected number of airplanes waiting to land and expected waiting time. What is the probability that the waiting will be more than 5 minutes? **(6 marks)**

b. Explain Markovian Birth Death process and obtain the expressions for steady state probabilities. **(8 marks)**

OR

18. a. A tax consulting firm has 3 counters in its office to receive people who have problems concerning their income, wealth and sales taxes. On the averages 48 persons arrive in an 8 hr day. Each tax advisor spends 15 mins on the average on an arrival. If the arrivals are Poisson distributed and service times are according to exponential distribution, find (i) the average number of customers in the system. (ii) the average number of customers waiting to be serviced. (iii) the average time a customer spends in the system. **(6 marks)**

b. Derive Erlang B formula. **(8 marks)**

19. a. What is Pollaczek-Khinchin formula? Derive the expression. **(8 marks)**

b. Consider a closed Jackson network where the service time at each queue is independent of the number of customers at the queue. Suppose that for a given number of customers, the utilization factor of one of the queues, say queue I, is strictly larger than the utilization factors of the other queues. Show that as the number of customers increases, the proportion of time that a customer spends in queue I approaches unity. **(6 marks)**

OR

20. a. State and prove Jackson's theorem. **(8 marks)**

b. Write short notes on closed Jackson networks and cyclic queues. **(6 marks)**

Syllabus

Module 1: 9 hours
(Text-1: Relevant topics from chapter-3) Linear Programming I: Simplex Method – Applications of Linear Programming – Standard Form of a Linear Programming Problem – Geometry of Linear Programming Problems– Definitions and Theorems – Solution of a System of Linear Simultaneous Equations – Pivotal Reduction of a General System of Equations – Identifying an Optimal Point – Improving a Nonoptimal Basic Feasible Solution – Two Phases of the Simplex Method
Module 2 : 10 hours
(Text-1: Relevant topics from chapter-4) Linear Programming II: Duality in Linear Programming – Symmetric Primal–Dual Relations – General Primal–Dual Relations – Primal–Dual Relations When the Primal Is in Standard Form – Duality Theorems – Dual Simplex Method – Sensitivity or Postoptimality Analysis – Changes in the Right-Hand-Side Constants b_i – Changes in the Cost Coefficients c_j – Addition of New Variables – Changes in the Constraint Coefficients a_{ij} – Transportation Problem – Karmarkar's Interior Method – Statement of the Problem – Conversion of an LP Problem into the Required Form.
Module 3: 9 hours
(Text-2: Relevant topics from sections-8.1-8.5, 8.7, 10.5)

Random processes and classification, mean and autocorrelation, wide sense stationary (WSS) processes, autocorrelation and power spectral density of WSS processes and their properties, Poisson process-distribution of inter-arrival times, combination of independent Poisson processes(merging) and subdivision (splitting) of Poisson processes (results without proof)
Module 4: 9 hours
(Text-3: Chapter 1, Chapter 2 – Section 2.1 to 2.7)   Introduction - Measures of System Performance, Characteristics of Queueing Systems, Little's Law, Some General Results. Stochastic Processes - Poisson Process, Exponential Distribution, Discrete Time Markov Chains, Continuous Time Markov Chains. Simple MarkovianQueueing Models - Birth-Death Processes - Single-Server Queues (M/M/1) – Multi server Queues (M/M/c) - Choosing the Number of Servers, Queues with truncation (M/M/c/K), Erlang's loss formula (M/M/c/c), Queues with unlimited service
Module 5 : 8 hours
(Text-3: Chapter 2- Section 2.8 to 2.12, Chapter 4- Section 4.1 to 4.4)   **(M/G/1 Queue – Text-4 : Chapter 3, Section 3.5)**   Finite Source Queues, State Dependent Service, Queues with Impatience, Transient Period, Busy Period Analysis, M/G/1 Queue, Series Queues, Open Jackson Networks, Closed Jackson Networks, Cyclic Queues

Text Books

1. Singiresu S. Rao, "Engineering Optimization: Theory and Practice, 4th Edition", Wiley 2009
2. Oliver C. Ibe, "Fundamentals of Applied Probability and Random Processes (Second Edition)", Academic Press, 2014
3. John F. Shortle, James M. Thompson, Donald Gross, Carl M. Harris, "Fundamentals of Queueing Theory, 5th Edition", Wiley 2018
4. Dimitri P. Bertsekas and Robert G. Gallager, "Data Networks," (2nd edition) Prentice Hall, 1992, ISBN 0132009161

Reference Books

1. Geoffrey R. Grimmett, David R. Stirzaker, Probability and Random Processes, Oxford University Press, USA; 3 edition, 2001.
2. Douglas C. Montgomery and George C. Runger, Applied Statistics and Probability for Engineers, Wiley, 2006.

3. Hamdy A. Taha, "Operations Research: An Introduction", 8th Edition, Pearson Education (2008).

4. G. V. Reklaitis, A. Ravindran, K. M. Ragsdell, "Engineering Optimization: Methods and Applications", Wiley (2006).

5. Leonard Kleinrock, "Queueing Systems. Volume 1: Theory", Wiley-Interscience, 1975.

6. Leonard Kleinrock, "Computer Applications, Volume 2, Queueing Systems", Wiley-Interscience. 1975.

7. Karlin, K. and Taylor, H. M.," A First Course in Stochastic Processes", Academic Press. 1975

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**MODULE 1**	**9 hours**
1.1	Linear Programming I: Simplex Method – Applications of Linear Programming – Standard Form of a Linear Programming Problem	2 Hrs
1.2	– Geometry of Linear Programming Problems– Definitions and Theorems	2 Hrs
1.3	Solution of a System of Linear Simultaneous Equations – Pivotal Reduction of a General System of Equations	2 Hrs
1.4	Identifying an Optimal Point	1 Hrs
1.5	Improving a Nonoptimal Basic Feasible Solution – Two Phases of the Simplex Method	2Hrs
2	**MODULE 2**	**10 hours**
2.1	Duality in Linear Programming – Symmetric Primal–Dual Relations – General Primal–Dual Relations – Primal–Dual Relations When the Primal Is in Standard Form – Duality Theorems – Dual Simplex Method	3 Hrs
2.2	Sensitivity or Postoptimality Analysis – Changes in the Right-Hand-Side Constants b_i – Changes in the Cost Coefficients c_j – Addition of New Variables – Changes in the Constraint Coefficients a_{ij}	3 Hrs
2.3	Transportation Problem – Karmarkar's Interior Method – Statement of the Problem – Conversion of an LP Problem into the Required Form	4 Hrs
3	**MODULE 3**	**9 hours**
3.1	Random processes and classification, mean and autocorrelation,	3 Hrs

3.2	wide sense stationary (WSS) processes, autocorrelation and power spectral density of WSS processes and their properties,	3 Hrs
3.3	Poisson process-distribution of inter-arrival times, combination of independent Poisson processes(merging) and subdivision (splitting) of Poisson processes (results without proof)	3 Hrs
4	**MODULE 4**	**9 hours**
4.1	Introduction - Measures of System Performance, Characteristics of Queueing Systems, Little's Law, Some General Results.	2 Hrs
4.2	Stochastic Processes - Poisson Process, Exponential Distribution, Discrete Time Markov Chains, Continuous Time Markov Chains.	3 Hr
4.3	Simple MarkovianQueueing Models - Birth-Death Processes - Single-Server Queues (M/M/1) – Multi server Queues (M/M/c) -	2 Hrs
4.4	Choosing the Number of Servers, Queues with truncation (M/M/c/K), Erlang's loss formula (M/M/c/c), Queues with unlimited service	2 Hrs
5	**MODULE 5**	**8 hours**
5.1	Finite Source Queues, State Dependent Service	2 Hr
5.2	Queues with Impatience, Transient Period, Busy Period Analysis	3 Hr
5.3	M/G/1 Queue, Series Queues, Open Jackson Networks, Closed Jackson Networks, Cyclic Queues	3 Hrs

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT294	NUMBER THEORY	VAC	3	1	0	4

Preamble:

Number theory doesn’t suffer too much abstraction and the consequent difficulty in conceptual understanding. Hence it is an ideal topic which acts like an essential bridge or tool from Mathematics to Information Technology. Important topics such as congruence, divisibility, Chinese remainder theorem, Classical results in Number theory, Application to ciphers are included in this course. Enthusiastic students will be able to acquire knowledge to read and enjoy their own more applications of Number theory.

Prerequisite: Linear Algebra and Calculus

Course Outcomes: After the completion of the course the student will be able to

CO No	Course Outcome(CO)	Bloom’s Category
CO 1	Examine results involving divisibility, greatest common divisor, Least Common multiple and a few applications	Apply
CO 2	Demonstrate theory and methods to solve Linear Difference Equations	Understand
CO 3	Summarize theory of congruence	Understand
CO 4	Solve linear congruent equations	Apply
CO 5	Illustrate three classical theorems of Number theory and Apply number theory to ciphers.	Apply

Mapping of course outcomes with program outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3	2	3	-	1	-	-	-	-	1	2
CO 2	3	3	2	2	-	1	2	-	-	-	1	2
CO 3	3	3	2	2	-	-	2	-	-	-	1	2
CO 4	2	2	2	2	-	1	-	-	-	-	1	2
CO 5	2	2	1	2	2	1	-	-	1	-	1	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	20	50
Apply	20	20	40

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. State Division algorithm.

2. List the first 4 Catalan numbers.

3. Distinguish between Fibinacci and Fermat numbers.

Course Outcome 2 (CO2)

1. State Fundamental theorem of Arithmetic

2. What is 73 congruent modulo 8.

3. State Chinese Remainder Theorem

Course Outcome 3(CO3):

1. State Fermat's little theorem
2. List first 3 Mersenne primes
3. Define Euler's Phi function

Course Outcome 4 (CO4):

1. Describe any two primality tests
2. Find primitive roots of 23.
3. State Lagrange's theorem

Course Outcome 5 (CO5):

1. Define Affine cipher
2. Define Hill ciphers
3. What is RSA crypto system.

Model Question paper

Course Code: IIT294

Course Name: NUMBER THEORY

Max.Marks:100 **Duration: 3 Hours**

PART A

Answer all questions. Each question carries 3 marks

1. Define polygonal numbers with examples using recurrence relation.
2. Express 3ABCsixteen in base ten.
3. Prove that any two consecutive Fibanacci numbers are relatively prime.
4. Evaluate (2076; 1776).
5. State Fermat's Little theorem.
6. Compute $\sum_{d|n} \varphi(d)$ for n = 12.
7. Verify that 2 is a primitive root modulo 9.
8. Explain perfect numbers and Mersenne primes with example
9. Using the exponentiation modulus p = 3037 and the enciphering key e = 31, encipher the message 'ALL IS WELL '.
10. Briefly explain RSA-crypto system.

PART B

Answer one full question from each module. Each full question carries 14 marks

MODULE 1

11. (a) Find the number of positive integer less than 3076 which are

(i) Divisible by 19 (ii) Not divisible by 24 (iii) Divisible by 17

(b) Prove that there are infinitely many primes

12. (a) Find a formula for $\sum_{i=1}^{n} F_i$

(b) Show that $641 \mid f_5$

MODULE 2

13. (a) Find the number of trailing zeros in 234!

(b) Solve $12x \equiv 18 (\mathrm{mod}\, 15)$

14. (a) Solve for x such that $x \equiv 1 (\mathrm{mod}\, 3)$; $x \equiv 4 (\mathrm{mod}\, 5)$; $x \equiv 6 (\mathrm{mod}\, 7)$

(b) Find the canonical decomposition and positive factors of 2520

MODULE 3

15. (a) State and prove Wilson's theorem

(b) Determine if there exist a positive integer f(m) such that $a^{f(m)} \equiv 1 (\mathrm{mod}\, m)$ for $m = 12$.

16. (a) Find the number of positive integers less than 500 and relative prime to 500. Also find the number and sum of positive divisors of 500.

(b) Find the reminder when 24^{1947} is divided by 17.

MODULE 4

17. (a) Compute $\mathrm{ord}_{21}5$

(b) Find the incongruent primitive roots modulo 19.

18. (a) State Lucas' theorem and verify that 823 is a prime using Lucas' theorem. (Take x= 2)

(b) Solve $8x^5 \equiv 3 (\mathrm{mod}\, 13)$

MODULE 5

19. (a) Using the matrix A = $A = \begin{bmatrix} 3 & 2 & 6 \\ 5 & 7 & 11 \\ 13 & 4 & 11 \end{bmatrix}$, encrypt the English proverb "A PROVERB IS THE CHILD OF EXPERIENCE ".

(b) Using the RSA enciphering modulus n = 2773 and the enciphering key e = 21, encrypt the message "SILENCE IS GOLDEN".

20. (a) Decrypt the cipher text message 0010 0325 2015 2693 2113 2398 2031 1857 that was created using the RSA enciphering key (e; n) = (21; 2773).

(b) Draw a block diagram for conventional cryptosystem and explain the terms.

Syllabus

Module 1 (9 hours)
Polygonal numbers – Pyramidal numbers – Catalan numbers – Division algorithm – Base b representations – Number patterns – Prime and composite numbers – Fibonacci and Lucas numbers – Fermat numbers
Module 2 (9 hours)
Greatest common divisor – Euclidean algorithm – Fundamental theorem of arithmetic – Least common multiple – Linear Diaophantine Equations – Congruences – Linear congruences – divisibility tests – Modular designs – Check digits – Chinese remainder theorem – General Linear systems - 2×2 Linear systems
Module 3 (11 hours)
Wilson's theorem – Fermat's little theorem – Euler's theorem – Euler's Phi function – Tau and Sigma function – Perfect numbers – Mersenne Primes
Module 4 (8 hours)
Order of a positive integer – Primality tests – Primitive roots of primes – Composites with primitive roots – The algebra of indices
Module 5 (8 hours)
Affine ciphers - Hill ciphers - Exponentiation ciphers – RSA Crypto system

Text Book

1. Thomas Koshy , "Elementary Number Theory with Applications (2/e)", Elsever Academic Press, 2007, ISBN: 978-0-12-372487-8.

Reference Books

1. David M Burton, "Elementary Number Theory (7/e)", McGraw Hill, 2011, ISBN : 978-0-07-338314-9

2. Gareth A Jones and J Mary Jones , "Elementary Number Theory", Springer Undergraduate Mathematics series, 1998, ISBN : 978-3-540-76197-6

3. Kenneth H Rosen, "Elementary Number Theory" (6/e)", Pearson Education, 2018, ISBN: 9780134310053

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Module 1**	**9 Hours**
1.1	Polygonal numbers – Pyramidal numbers	1
1.2	Catalan numbers	1
1.3	Division algorithm	1
1.4	Base b representations – Number patterns	1
1.5	Prime and composite numbers	1
1.6	Fibonacci and Lucas numbers	3
1.7	Fermat numbers	1
2	**Module 2**	**9 Hours**
2.1	Greatest common divisor – Euclidean algorithm	1
2.2	Fundamental theorem of arithmetic - Least common multiple	1
2.3	Linear Diaophantine Equations	1
2.4	Congruences – Linear congruences	2
2.5	divisibility tests – Modular designs – Check digits	1
2.6	Chinese remainder theorem	2
2.7	General Linear systems - 2×2 Linear systems	1
3	**Module 3**	**11 Hours**
3.1	Wilson's theorem – Fermat's little theorem	4
3.2	Euler's theorem – Euler's Phi function	4
3.3	Tau and Sigma function	2
3.4	Perfect numbers – Mersenne Primes	1
4	**Module 4**	**8 Hours**
4.1	Order of a positive integer	2
4.2	Primality tests	3
4.3	Primitive roots of primes – Composites with primitive roots	2
4.4	The algebra of indices	1
5	**Module 5**	**8 Hours**
5.1	Affine ciphers	2
5.2	Hill ciphers	2
5.3	Exponentiation ciphers	2
5.4	RSA Crypto system	2

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT296	MICROPROCESSOR AND MICROCONTROLLER PROGRAMMING	VAC	2	1	1	4

Preamble: Microprocessor and Micro controller programming course is intended to deliver students the concepts of Microprocessors and Micro-controllers. It also helps them to learn how to write an 8051 program assembly language and also in C programming language. Introduction to Interfacing of micro-controllers, its use and applications are also covered in the syllabus.

Prerequisite: C programming

Course Outcomes: After the completion of the course the student will be able to

CO No	Course Outcome(CO)	Bloom's Category
-CO 1	Describe the basic architectures of microprocessor based systems	Understand
CO 2	Develop a simple assembly program for a 8086 microprocessor	Apply
CO 3	Design a basic 8051 program in Assembly language	Apply
CO 4	Simulate assembly programs using simulation tools and design 8051 programs in C programming language	Apply
CO 5	Utilize various interfacing techniques of micro-controllers	Apply

Mapping of course outcomes with program outcomes

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	-	-	-	-	-	-	1	-
CO2	3	3	3	-	2	-	-	-	-	-	1	3
CO3	3	3	3	-	2	-	-	-	-	-	1	3
CO4	3	3	3	-	3	-	-	-	-	-	1	3
CO5	3	-	-	-	3	3	-	-	3	2	3	3

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Test		End Semester Examination
	1	2	
Remember	15	15	30
Understand	15	15	30
Apply	20	20	40
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):
1. Differentiate CISC and RISC processors
2. With the help of an example show the set and reset conditions of CY, AC and P flag of 8051 micro controller

Course Outcome 2 (CO2):
1. Write an 8086 program to sort N numbers.

2. Write an 8086 program to find the sum of N numbers

Course Outcome 3 (CO3):
1. Write an 8051 program to find the factorial of a given number.
2.Write an 8051 assembly code to generate look up table for Fibonacci numbers.

Course Outcome 4 (CO4):
1. Write an 8051 C program to toggle bits of P1 ports continuously with 250 ms.
2. Write an 8051 C program to convert 11111101(FDH) to decimal and display the digits on port p0, p1 and p2.

Course Outcome 5 (CO5):
1. Explain the function of pins of 9 pin RS 232 connector
2. Illustrate different modes of operations of 8255 with respect to control words.

Model Question Paper

PART A **(10*3=30)**

(Each question carries 3 Marks)

1.Compare microprocessor, microcomputer and micro-controller

2.Differentiate PIC and AVR Micro-controllers

3.List the instructions which are used for memory operation in 8088 microprocessor.

4.Explain the purpose of the following signals in 8086

(i) READY (ii) HOLD

5.Write a program to do data conversions from HEX to ASCII in 8051 assembly code.

6.Explain MOV, MOVC, MOVX instructions of 8051 Micro-controller.

7. Discuss any three data types of 8051 C.

8. Illustrate with an example how arrays are used in 8051 C programs.

9.Describe different modes of operation of the following peripheral ICs:

i) 8255 ii) 8257

10.Write the Control Word Format in 8255.

PART B **(5*14=70)**

(Each full question carries 14 marks)

11. a) ExplainVan Neumann and Harvard Architecture. (6 marks)

b) Draw the memory map and briefly explain the memory organization for 128 byte internal RAM of 8051 micro-controller. (8 marks)

OR

12.a) Explain the functions of ports in 8051 micro-controller. How can P1 be used as both output and input port? (5 marks)

b) Draw the pin diagram of 8051 micro-controller and explain the function of each pin.(9 marks)

13. a) Compare the architectural features of 8086 and 8088 processors. (7 marks)

b) An array of 10 numbers is stored in the internal data RAM starting from location 30 H. Write an assembly language program to sort the array in ascending order starting from location 40 H. (7 marks)

OR

14. a) Draw the architectural diagram of 8086 microprocessor and explain. (9 marks)

b) Write an assembly program to add N numbers. (5 marks)

15. a) Assuming crystal frequency of 11.0592 MHz, write an 8051 assembly program to generate a square wave of 50Hz frequency on pin P2.0 of 8051 Micro-controller.

[Hint: Interfacing is not to be considered here for generating square wave] (6 marks)

b) Discuss the addressing modes of 8051 Instruction set. (8 marks)

OR

16. a) Write an 8051 based assembly language program to perform addition of two 2x2 matrices. (7 marks)

b) Write an 8051 based assembly language program to covert a hexadecimal number to a BCD. Number. (7 marks)

17. a) Write an 8051 C program to generate the 500us time delay using T1M2(timer1 and mode2). (7 marks)

b) Write an 8051 C program to read the P1.0 and P1.1 bits and then issue an ASCII character to P0 based on the following conditions. That is if the data from P1.1 and P1.0 is 00, send ‘0’ if 01 send ‘1’, if 10 send ‘2’ . (7 marks)

OR

18. a) Write an 8051 C to get a byte of data from port P0. If the data is greater than 100 send it to P1 otherwise send it to P2. (7 marks)

b) Write an 8051 C program to toggle all the bits of P0, PI, and P2 continuously with a 250 ms delay.’ Use the sfr keyword to declare the port addresses. (7 marks)

19. a) Explain the architecture of programmable interrupt controller 8259. (7 marks)

b)Explain the interfacing of 8 bit ADC using 8051 micro-controller. (7 marks)

OR

20. a) Explain Keyboard Display controller 8279. (5 marks)

b) Give the advantage of using 8279 for keyboard/display interface? What are scan lines used for? Explain (i) Encoded Scan Mode and (ii) Decoded scan mode. (9 marks)

Syllabus

Module 1:(8 hours)
Microprocessor Based Systems: Digital Computer, Microprocessor, Microcomputer, Micro-controller, Van Neumann and Harvard Architecture, CISC and RISC Processors; **Micro-controllers:** Historical background; organization and architectural features of micro-controller 8051 ,Introduction to AVR and PIC micro-controllers
Module 2:(10 hours)
Organization and architectural features of microprocessor 8086,Introduction to 8088 microprocessors, **Introduction of assembly language program-**Complete 8086 instruction set and Basic programs in assembly language for 8086 should be covered & asked in the exam.
Module 3:(9 hours)
8051 programming in Assembly language : Introduction to instruction set: instruction format, addressing modes of 8051, Data transfer instructions, I/O Port programming, Arithmetic and Logical instructions, Bit level instructions, Branching instructions , Concept of stack, subroutine and related instructions, writing programs (like time delay using loop, data conversions HEX to ASCII, BCD to ASCII, use of look up table etc) in assembly language 8051
Module 4:(10 hours)
Introduction to Program Development Environment (IDE)Tools: Introduction to a simulator: Edge Simulator- Edsim - Programming &Testing using IDE. **8051 Programming in C:**Data types, programming for time delay, I/O programming, Logic operations, Control statements and loops, Functions and Arrays in embedded C, Data conversion programs in 8051 C, Accessing code of ROM space & Data serialization using 8051 C.
Module 5:(8 hours)

Interfacing of micro-controllers: Interfacing of memory devices - data transfer techniques and I/O ports (8255); keyboard and display devices(8279) - programmable interrupt and DMA controllers (8257) - sensors, transducers, actuators, A/D and D/A Converters - standard interfaces - RS232, USB, Simple interfacing programs using 8051- (Group Mini projects can be given. Can be evaluated as Assignments. Interfacing programs need not be asked for exams)

Text Books

1. R. Gaonkar, Microprocessor Architecture, Programming, and Applications with the 8085, 6/e,Penram International Publishers.
2. A. Pal, Microprocessors: Principles and Applications, 1990,Tata McGraw-Hill.
3. K. J. Ayala, The 8051 Microcontroller Architecture, Programming and Applications, 2/e, Penram International Publishers.
4. Mazidi ,Mazidi, McKinlay, Microcontroller and Embedded Systems, 2/e, Pearson Education.
5. R. Kapadia, 8051 Microcontroller and Embedded Systems, 1/e, Jaico Publishing House.
6. Abubeker K M, 80C51 µC - Embedded C & ALP Programming-ISBN-978-1648921216, Notionpress publications, Singapore, 2020 , first edition.

Course Content and Course Schedule

Module 1: Microprocessor Based Systems and Micro controllers		**8 hours**
1.1	Introduction to Digital Computer and Microprocessor	1 hour
1.2	Introduction to Microcomputer, Micro-controller	1 hour
1.3	Introduction to Van Neumann and Harvard Architecture	1 hour
1.4	Introduction to CISC and RISC Processors,	1 hour
1.5	Micro-controllers: Historical background	1 hour
1.6	Architecture of micro-controller 8051	1 hour
1.7	Register and memory organization of 8051	1 hour

1.8	Introduction to AVR, PIC and ARM micro controllers	1 hour
Module 2: Architecture of 8086 & Introduction of assembly language program		**10 hours**
2.1	Architecture of microprocessor 8086	1 hour
2.2	Memory Organization of 8086	1 hour
2.3	Introduction to 8088 microprocessor	1 hour
2.4	Introduction of assembly language program	1 hour
2.5	Instruction Types and Addressing modes	1 hour
2.6	Data transfer instructions, I/O Port programming, Arithmetic instructions	1 hour
2.7	Logical and Bit level instructions, Branching instructions	1 hour
2.8	Introducing Sample assembly language programs for 8086 (Square, Square Root & Cube Root of a Number, Factorial of an 8-bit Number, Generation of Fibonacci Series, HCF and LCM of Two Numbers, Bubble Sorting, Largest and Smallest Number of an Array, Code conversion – HEX to Decimal & ASCII to HEX, HEX to ASCII, BCD to ASCII and Matrix Addition. These programs can be done during tutorial/practical)	1 hour
2.9	Writing/Doing the above set of programs in assembly language for 8086 - Tutorial/Practical	1 hour
2.10	Writing/Doing the above set of programs in assembly language for 8086 - Tutorial/Practical	1 hour
Module 3: 8051 programming in Assembly language		**9 hours**
3.1	Introduction to instruction types and instruction format	1 hour
3.2	Introduction to Instruction sets and addressing modes of 8051	1 hour
3.3	Data transfer instructions, I/O Port programming, Arithmetic instructions	1 hour
3.4	Logical and Bit level instructions	1 hour
3.5	Branching instructions (Jump and loop Jump and call)	1 hour
3.6	Writing /Doing Programs in 8051-Tutorial/Practical	1 hour
3.7	Concept of stack, subroutine and related instructions	1 hour

3.8	Writing programs (like time delay using loop, data conversions HEX to ASCII, BCD to ASCII, use of look up table etc)in assembly language 8051	1 hour
3.9	Writing /Doing Programs in 8051-Tutorial/Practical	1 hour
Module 4: **Introduction to Tools & 8051 Programming in C**		**10 hours**
4.1	Introduction to Program Development Tools (IDE): Concept of IDE, Editor, Assembler, Compiler, Linker, Simulator, Debugger and assembler directives.	1 hour
4.2	Introduction to a simulator: Edge Simulator- Edsim - Programming using Simulator-Testing programs using IDE	1 hour
4.3	Writing /Doing Programs in 8051 using Tools -Practical in Lab	1 hour
4.4	8051 Programming in C: Data types in 8051 C	1 hour
4.5	Programming for time delay, I/O programming in 8051 C	1 hour
4.6	Logic operations in 8051 C, Control statements and loops in embedded C	1 hour
4.7	Doing Programs in 8051 using Tools - Practical in lab	1 hour
4.8	Functions and Arrays in embedded C, Data conversion programs in 8051 C	1 hour
4.9	Accessing code ROM space using 8051 C, Data serialization using 8051 C	1 hour
4.10	Doing Programs in 8051in C -Practical in lab	1 hour
Module 5: **Interfacing of micro-controllers**		**8 hours**
5.1	Introduction to Interfacing of micro-controllers: Use and Applications	1 hour
5.2	Interfacing of memory devices; data transfer techniques and I/O ports (8255)	1 hour
5.3	Interfacing of keyboard and display devices(8279)	1 hour
5.4	Programmable interrupt and DMA controllers (8257)	1 hour
5.5	Doing Interfacing programs in 8051 using Tools - Practical in lab	1 hour
5.6	Interfacing of sensors, transducers, actuators	1 hour
5.7	A/D and D/A Converters - standard interfaces - RS232, USB	1 hour
5.8	Doing Interfacing programs in 8051 using Tools - Practical in lab	1 hour

SEMESTER V

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT301	**WEB APPLICATION DEVELOPMENT**	**PCC**	**3**	**1**	**0**	**4**

Preamble: Web Application Development course is intended to deliver the elementary concepts of Web Application Development with HTML, CSS, JavaScript, JQuery, Node JS and MongoDB thereby equipping them to develop real time web applications.

Prerequisite: Basics of programming

Course Outcome (CO): After completion of the course, the student will be able to

CO No.	Course Outcomes	Bloom's Category
CO1	Identify HTML5 elements in webpages	Level 2: Understand
CO2	Implement Cascading Stylesheet to add style in HTML pages	Level 3: Apply
CO3	Apply JavaScript to add functionality to web pages	Level 3: Apply
CO4	Use Ajax & JQuery to enhance the functioning of web pages	Level 3: Apply
CO5	Develop web applications with HTML, CSS, JavaScript, Node JS and MongoDB	Level 3:Apply

Mapping of Course Outcomes with Program Outcomes 3/2/1: High/Medium/Low

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	2	2		2							
CO2	1	2	2		2							
CO3	2	2	1		2							
CO4	2	2	1		3							1
CO5	2	3	1	1	3							2

Assessment Pattern

Bloom's Category	Continuous Assessment Tests (Marks)		End Semester Examination (Marks)
	1	2	
Remember			
Understand	30	20	40
Apply	20	30	60
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Test (2 numbers) : 25 marks

Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

CO1: Differentiate ordered list and unordered list with example.

CO2: Create a navigation bar in the format shown below using Cascading Stylesheet and HTML:

CO3: Demonstrate insertion of new list item in an HTML page utilizing DOM methods.

CO4: Use Ajax & JQuery to enhance the functioning of web pages

CO5: Create a web application for library with HTML, CSS, JavaScript and Node. The features needed in the website are:

1. Insert Book
2. Delete Book
3. Checkout Book

Model Question Paper

Course Code: ITT301

Course Name: WEB APPLICATION DEVELOPMENT

Max.Marks:100 Duration: 3 Hours

PART A **(10*3=30)**

(Each question carries 3 marks)

1. What is the use of href? Give example.
2. Illustrate the usage of alt attribute in an image tag.
3. What are Cascading Style Sheets?
4. Differentiate block and inline elements.
5. What is Document Object Model?
6. Illustrate how JavaScript makes webpages more interactive.
7. Differentiate let, var and const in JavaScript.
8. Illustrate how ajax works?
9. What are the different operations involved in accessing a web page?
10. Explain different features of node.js.

PART B **(5*14=70)**

11. Explain table tag and create the following table using table tag in HTML:

Branch	CGPA/Percentage		Salary
	UG	PG	
IT	105	12	500000
Others	200	225	400000

OR

12. Explain various concerns and operations involved in web design starting from ideation to hosting of a website.

13. Differentiate the concepts of inline, internal and external style sheets with examples.

OR

14. Illustrate layout and positioning elements in CSS with example.

15. Explain JavaScript: Objects: Math, String, Date, and document Object with example.

OR

16. What is events and explain event handling with example.

17. Explain callbacks, promise and async/await with example.

OR

18. What is Ajax, and explain loading JSON with Ajax.

19. Discuss CRUD operation with node express.

OR

20. Explain steps involved in building a node express app with MongoDB.

Syllabus

MODULE 1: INTRODUCTION TO WEB DESIGNING	8 Hrs
Web Design Basics: Who is the Site For?, Why People Visit your Website?, What Information Your Visitors Need?, Site maps, wireframes, Getting your message across using design, Visual hierarchy, grouping and similarity, Designing Navigation, Search Engine Optimization (SEO), Analytics, Domain Names & Hosting, Ftp & Third party tools **HTML5:** Introduction to HTML5, Basic Structure for HTML, Basic HTML tags-Headings, Linking, Images, Special Characters and Horizontal Rules, Lists, Tables, Forms, Internal Linking, meta elements, New HTML5 Form input Types, input and data list elements, autocomplete Attribute, Page-Structure Elements	
MODULE 2: STYLE WITH CSS	**9 Hrs**
Introduction to CSS: Introduction to CSS, Block and Inline Elements, Inline Styles, using internal CSS, using external CSS, How CSS rules cascade, inheritance, why use external style sheets? **CSS3 Basics:** CSS selectors, *color:* foreground color, background color, contrast, opacity; *text:* Typeface terminology, Specifying Typefaces, font-size, font-weight, font-style, text-transform, text-decoration, line-height, letter-spacing, word-spacing, text-align, vertical-align, text-indent, text-shadow; responding to users; *box:* box dimensions, limiting width, limiting height, overflow; *border margin and padding*, centering content, change inline/blocks, hiding boxes, box shadows, rounded corners; *list tables and forms:* list-style, table properties, styling forms, styling text input **Layout and positioning:** *layout:* key concepts in positioning elements, *controlling the position of elements:* relative positioning, absolute positioning, fixed positioning, z-index, float, clear, creating multi column layout with float, fixed width layout, liquid layout, layout grids, *Images:* controlling size of images in CSS, aligning images using CSS, centering	

images using CSS, background images, gradients, Media Queries	
MODULE 3: INTRODUCTION TO JAVASCRIPT	**9 Hrs**
JavaScript: How JavaScript makes the webpages more interactive, examples of JavaScript in browser, *Basic JavaScript instructions:* statements, comments, variable, data types, arrays, expressions, operators; *functions methods and objects:* function, anonymous function, variable scope, object, this, arrays are objects, browser object model, document object model, *Global objects:* string, number, math, date. **Decision making and Loops:** *decision making:* if statement, if…else statement, switch statement, *loops:*key loop concepts, for loops, while loops, do while loops; **DOM:** Document Object Model (DOM), the DOM tree as a model of a web page, working with DOM tree, accessing elements, nodelists, selecting elements: using class attribute, tag name, CSS selectors; repeating actions for an entire nodelist, looping through a nodelist, traversing the DOM, adding or removing html content, update text and markup, adding/removing elements **Event handling:** different event types and ways to bind an event to an element: using DOM event handlers, using event listeners, using parameters with event listeners; the event object, event delegation, user interface events, event bubbling	
Module 4: JAVASCRIPT ADVANCED	**10 Hrs**
ECMA Script: ECMA Script versions, ES5 Features, ES6 introduction, Var Declarations and Hoisting, let declaration, Constant declaration, function with default parameter values, default parameter expressions, unnamed parameters, the spread operator, arrow functions, object destructuring, array destructuring, sets and maps, Array.find(), Array.findIndex(), template strings, Javascript classes, callbacks, promises, async/await **AJAX:** What is Ajax? , Why use Ajax?, How Ajax works?, Handling Ajax request and response, data formats: XML, JSON; Working with JSON data, Loading HTML with Ajax, Loading XML with Ajax, Loading JSON with Ajax, working with data from other servers **JQuery :** What is JQuery ?, A basic JQuery example, Why use JQuery ?, finding elements, JQuery selection, getting element content, updating elements, changing content, inserting elements, adding new content, getting and setting attributes, getting and setting CSS properties, using .each(), events, event object, effects, animating CSS properties, using animation, traversing the DOM, working with forms, JavaScript libraries, JQuery and	

Module 5: BACK END DEVELOPMENT	9 Hrs
Ajax	
Web Servers: Introduction, HTTP Transactions, Multitier Application Architecture, Client-Side Scripting versus Server-Side Scripting, Accessing Web Servers. **Server Side Scripting with Node.js:** Getting to know node, node.js changed JavaScript forever, features of node, when to use and not use node, asynchronous callbacks, the NoSql movement, node and MongoDB in the wild, Hello World in Node, package.json, modules, *Built-in Modules:* FS Module, HTTP Module, Events; Node Package Manager(npm), web server using http, node.js with express, middleware, routing in express, CRUD operations in express, web server using express, making it live on Heroku **Node.js with MongoDB**: basics of MongoDB, MongoDB CRUD Operations, Building a data model with MongoDB and Mongoose, Defining simple mongoose schemas, build node express app with MongoDB	

Text Books

1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, “Internet and World Wide Web How To Program”, 5/E, Pearson Education, 2012.

2. Jon Duckett , “HTML and CSS: Design and Build Websites”, Wiley

3. Jon Duckett , “JavaScript and JQuery : Interactive Front–End Web Development”, Wiley

4. Nicholas C. Zakas, “Understanding ECMAScript 6: The Definitive Guide for JavaScript Developers”

Reference Books

1. Alex Young, Marc Harter, “Node js in practice”, Manning.

2. Json Krol , “Web Development with MongoDB and node js”, Packt

3. Krishna Rungta , “Node JS: learn in one day

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures (in hours)
1	**Introduction to web designing**	**8**
1.1	**Web Design Basics:** Who is the Site For?, Why People Visit your Website, What Information Your Visitors Need?, Site maps, wireframes, Getting your message across using design, Visual hierarchy, grouping and similarity, Designing Navigation, Search Engine Optimization (SEO), Analytics, Domain Names & Hosting, Ftp & Third party tools	2
1.2	**HTML5:** Introduction to HTML5, Basic Structure for HTML, Basic HTML tags-Headings, Linking, Images, Special Characters and Horizontal Rules	2
1.3	Lists, Tables, Forms, Internal Linking, meta elements, New HTML5 Form input Types	2
1.4	input and data list elements, autocomplete Attribute, Page-Structure Elements	2
2	**Style with CSS**	**9**
2.1	**Introduction To CSS:** Introduction to CSS, Block and Inline Elements, Inline Styles, Using internal CSS, Using external CSS, How CSS rules cascade, inheritance, why use external style sheets?	2
2.2	**CSS3 Basics:** CSS selectors, *color:* foreground color, background color, contrast, opacity; *text:* Typeface terminology, Specifying Typefaces, font-size, font-weight, font-style, text-transform, text-decoration, line-height, letter-spacing, word-spacing, text-align, vertical-align, text-indent, text-shadow; responding to users; *box:* box dimensions, limiting width, limiting height, overflow; *border margin and padding*, centering content, change inline/blocks, hiding boxes, box shadows, rounded corners	2
2.3	*list tables and forms:* list-style, table properties, styling forms, styling text input	1
2.4	**Layout and positioning:** *layout:* key concepts in positioning elements, *controlling the position of elements:* relative positioning, absolute positioning, fixed positioning, z-index, float, clear, creating multi column layout with float, fixed width layout, liquid layout, layout grids,	2

2.5	*Images:* controlling size of images in CSS, aligning images using CSS, centering images using CSS, background images, gradients, Media Queries	2
3	**Introduction To JavaScript**	**9**
3.1	**JavaScript:** How JavaScript makes the webpages more interactive, examples of JavaScript in browser, *Basic JavaScript instructions:* statements, comments, variable, data types, arrays, expressions, operators; *functions methods and objects:* function, anonymous function, variable scope, object, this, arrays are objects, browser object model, document object model, *Global objects:* string, number, math, date;	2
3.2	**Decision making and Loops:** *decision making:* if statement, if…else statement, switch statement, *loops:* key loop concepts, for loops, while loops, do while loops;	2
3.3	**DOM:** Document Object Model (DOM), the DOM tree as a model of a web page, working with DOM tree, accessing elements, nodelists, selecting elements: using class attribute, tag name, CSS selectors; repeating actions for an entire nodelist, looping through a nodelist,	2
3.4	traversing the DOM, adding or removing html content, update text and markup, adding/removing elements	1
3.5	**Event handling:** different event types, three ways to bind an event to an element, using DOM event handlers, using event listeners, using parameters with event listeners, the event object, event delegation, user interface events, event bubbling	2
4	**JavaScript Advanced**	**10**
4.1	**ECMA Script:** ECMA Script versions, ES5 Features, ES6 introduction, Var Declarations and Hoisting, let declaration, Constant declaration, function with default parameter values, default parameter expressions, unnamed parameters, the spread operator, arrow functions, object destructuring, array destructuring, sets and maps, Array.find, Array.findIndex, template strings	2
4.2	JavaScript classes, callbacks, promises, async/await	1
4.2	**AJAX:** What is Ajax?, Why use Ajax?, How Ajax works?, Handling Ajax request and response, data formats: XML, JSON; Working with JSON data, Loading HTML with Ajax,	2

4.3	Loading XML with Ajax, Loading JSON with Ajax, working with data from other servers	1
4.4	**JQUERY :** What is JQuery ?, A basic JQuery example, Why use JQuery ?, finding elements, JQuery selection, getting element content, updating elements, changing content, inserting elements, adding new content, getting and setting attributes	2
4.5	getting and setting CSS properties, using .each(), events, event object, effects, animating CSS properties, using animation, traversing the DOM, working with forms, JavaScript libraries, JQuery and Ajax	2
5	**Back End Development**	**9**
5.1	**Web Servers:** Introduction, HTTP Transactions, Multitier Application Architecture, Client-Side Scripting versus Server-Side Scripting, Accessing Web Servers.	2
5.2	**Server Side Scripting with Node.js:** Getting to know node, node.js changed JavaScript forever, features of node, when to use and not use node, asynchronous callbacks, the NoSql movement, node and MongoDB in the wild, Hello World in Node, package.json, modules,	2
5.3	*Built-in Modules:* FS Module, HTTP Module, Events; Node Package Manager(npm), web server using http, node.js with express, middleware, routing in express, CRUD operations in express, web server using express, making it live on Heroku	2
5.4	**Node.js with MongoDB**: basics of MongoDB, MongoDB CRUD Operations, Building a data model with MongoDB and Mongoose	2
5.5	Defining simple mongoose schemas, build node express app with MongoDB	1

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT303	OPERATING SYSTEM CONCEPTS	PCC	3	1	0	4

Preamble: Operating System Concepts is a graduate-level introductory course in operating systems. This course teaches the basic operating system functions, abstractions, mechanisms, and their implementations. The course is split into five modules: (1) Introduction to OS, (2) Process Management, (3) Process Synchronization, (4) Memory Management and (5) Storage Management.

Prerequisite: ITT204 Computer Organization

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category
CO 1	Explain the concepts and functionality of operating systems.	Level 2: Understand
CO 2	Describe the concepts of process management and process synchronization and apply them to solve problems.	Level 3: Apply
CO 3	Illustrate deadlock and deadlock – prevention and avoidance techniques.	Level 3: Apply
CO 4	Illustrate the memory management techniques.	Level 3: Apply
CO 5	Explain the file system and its implementation	Level 2: Understand
CO 6	Use the disk scheduling algorithms to solve problems.	Level 3: Apply

Mapping of course outcomes with program outcomes 3/2/1: High/Medium/Low

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	1	1									2
CO 2	3	3	3	2	1							2
CO 3	3	3	3	2	1							2
CO 4	3	3	3	2	1							2
CO 5	3	2	2	1								2
CO 6	3	3	3	2	1							2

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	5	5	10
Understand	20	20	40
Apply	25	25	50
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Test (2 numbers) : 25 marks

Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14marks.

Course Level Assessment Questions
Course Outcome 1 (CO1):
1. Describe the basic functions of operating system.

2. Illustrate the various types of operating system.

3. Explain the types of System Calls.

Course Outcome 2 (CO2):

1.Describe the process state with suitable diagram.

2. Consider the following set of processes with CPU burst given in seconds.

Process	CPU Burst
P1	20
P2	4
P3	6
P4	4

i. Draw the Gantt chart for FCFS and Round Robin (Time quantum=4s).

ii. What is the average waiting time for each of the scheduling algorithm?

3. Explain the fields in a process control block. What is the use of PCB in context switching?

Course Outcome 3 (CO3):

1. Explain deadlocks detection techniques.

2. Describe deadlock and necessary conditions for deadlocks.

3. Develop the program for Banker's algorithm.

4. Does a cycle in a resource allocation graph indicate a deadlock situation? Justify your answer.

5. Demonstrate the use of Peterson's solution to the critical section problem.

Course Outcome 4 (CO4):

1. Explain internal fragmentation and external fragmentation with suitable diagrams.

2. Illustrate paging and segmentation with suitable diagram.

3. Consider the following page reference string: 1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6. How many page faults would occur for the following replacement algorithms, assuming three frames? What happens when four frames are used? All frames are initially empty, so your first pages will all cost one fault each (i) LRU replacement (ii)FIFO.

Course Outcome 5 (CO5):

1. Illustrate the File Attributes & File Operations.

2. Identify the different File Access methods.

3. Illustrate the various Directory structures.

Course Outcome 6 (CO6):

1. Explain the various Disk scheduling algorithms.

2. Consider a disk containing 200 cylinders. At a certain point of time the disk head is at cylinder 55 and the disk queue contains request for I/O to blocks on cylinders 58, 39, 150, 180, 65, 75, 88, 110, 100,130. Find out the total head movement with respect to FCFS, SSTF, SCAN, C-SCAN and LOOK scheduling.

3. How would you select a disk scheduling algorithm?

Model Question paper

Course Code: ITT303
Course Name: Operating System Concepts

Max.Marks:100 **Duration: 3 Hours**

PART A

Answer all Questions. Each question carries 3 Marks

1. What is an operating system? State and explain the basic functions of operating system?

2. Differentiate between hard real-time systems and soft real-time systems. Give 2 examples of each.

3. Explain five state process models with a neat diagram.

4. Compare long term scheduling and short term scheduling.

5. What is a deadlock? What are the necessary conditions for a deadlock to occur?

6. Explain the concept of safe and unsafe state in the context of deadlock avoidance.

7. Differentiate between external fragmentation and internal fragmentation.

8. What is Translation Look Aside Buffer (Associative Memory)? What is the need for TLB?

9. List out the different operations that can be performed by a file system.

10. What is Direct memory access technique and how it is advantageous in performing I/O?

(10x3=30)

Part B

Answer any one Question from each module. Each question carries 14 Marks

11.	(a) Explain the various types of system calls with an example for each.	(10)
	(b) Explain the execution of a system call.	(4)
	OR	
12.	(a) Explain in detail about the OS structure.	(10)
	(b)What are shells? Give examples.	(4)
13.	(a) What is context switching? What are all the factors affecting context switching time?	(6)
	(b) Explain any two preemptive CPU scheduling algorithms with example.	(8)
	OR	
14.	(a) What are the functions of a dispatcher?	(6)
	(b) Explain the structure of PCB.	(8)
15.	(a) What are the strategies for recovering from deadlock? Write the merits and demerits of each.	(10)
	(b) Explain how dead lock can be prevented in a system.	(4)
	OR	

16. (a) Explain how resource allocation graphs can be used to detect deadlock. (6)

(b) Explain Bankers algorithm for dead lock avoidance with multiple resources of each type. (8)

17. (a) Describe the following memory allocation algorithms. (6)

i) First-fit
ii) Best–fit
iii)Worst –fit

(b) Explain paging memory management techniques. (8)

OR

18. (a) Explain how segmentation with paging is implemented. (8)

(b) What is virtual memory? How is it implemented? (6)

19. (a) How directories are implemented? (6)

(b) Explain the different types of directory structures. (8)

OR

20. (a) Describe various file access methods. (4)

(b) Illustrate the disk scheduling algorithms. (10)

(14x5=70)

Syllabus

Module 1 (7 hours)
Operating Systems: Introduction, Functions of OS, Types of OS (Batch, Multi programmed, Time-sharing and Real time systems) –System calls – System Programs — System structure (Simple structure, Layered approach, Microkernel system structure, Modules)– Kernel, Shell.
Module 2 (11 hours)
Process Management: Process concept, Process State, PCB, Operations on processes, Multithreading-Benefits. **Process Scheduling:** Basic concepts, Preemptive Scheduling, Dispatcher, Scheduling criteria, Scheduling Algorithms (FCFS, SJF, Priority scheduling, Round Robin Scheduling, Multi level queue scheduling, Multi level feedback queue scheduling). **Inter process communication** (Shared memory, message passing, pipes and socket).
Module 3 (11 hours)
Process Synchronization: Race Conditions - Critical Sections – Mutual Exclusion - Busy Waiting - Sleep and Wakeup - Semaphores – Monitors (Introduction). **Deadlocks**: Deadlock characteristics - conditions for deadlock - prevention – avoidance (Safe state, Resource –Allocation Graph, Banker's algorithm) - deadlock detection – recovery from dead lock.
Module 4 (10 hours)
Memory Management: Basics - Swapping -Memory Allocation (fixed partitions, variable partitions) Fragmentation - Paging - Segmentation - Virtual memory concepts – Demand paging - Page replacement algorithms (FIFO, Optimal, LRU) – Allocation of frames - Thrashing.
Module 5 (6 hours)
Storage Management: File System: Introduction, File concept – File Attributes, File Operations, File Types, File structure- File access methods (Sequential Access, Direct Access, Indexed Access)– File allocation methods (Contiguous, linked and indexed allocation), Directory structure (Single-Level, Two-Level, Tree-Structured, Acyclic Graph, General Graph)– Directory implementation (Linear list, Hash table). Disk Management: Introduction, Disk Scheduling algorithms (FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK)

Text Books

1. Andrew S. Tanenbaum and Herbert Bos, Modern Operating Systems, 4th edition, Pearson, 2015
2. A. Silberschatz, G.Gagne and P.Galvin, Operating System Concepts, 7th edition, AddisonWesley, 2004.

Reference Books

1. D M Dhamdhere, "*Operating Systems A Concept-based Approach*", Tata McGraw Hill, New Delhi, 2nd Edition, 2010.
2. William Stallings, Operating Systems, 6th Edition, Pearson, 2009.
3. Garry Nutt, "Operating Systems – A Modern perspective", Third Edition, Pearson Education

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Operating Systems**	**(7 hours)**
1.1	Introduction, Functions of OS	1 hour
1.2	Types of OS	2 hours
1.3	System calls	1 hour
1.4	System Programs	1 hour
1.5	System structure	1 hour
1.6	Kernel & Shell.	1 hour
2	**Process Management**	**(11 hours)**
2.1	Process concept, Process State, PCB	2 hours
2.2	Operations on processes	1 hour
2.3	Multithreading-Benefits.	1 hour
2.4	**Process Scheduling:** Basic concepts	1 hour
2.5	Pre-emptive Scheduling, Dispatcher	1 hour
2.6	Scheduling criteria	1 hour
2.7	Scheduling Algorithms	3 hours
2.8	Inter process Communication	1 hours
3	**Process Synchronization**	**(11 hours)**
3.1	Race Conditions - Critical Sections	1 hour
3.2	Mutual exclusion with busy waiting	2 hours
3.3	Sleep and Wakeup	1 hour
3.4	Semaphores, Monitors(introduction)	2 hours
3.5	**Deadlocks**: Deadlock characteristics, conditions for deadlock	1 hour
3.6	Deadlock prevention	1 hour
3.7	Deadlock avoidance	2 hours

3.8	Deadlock detection & recovery from dead lock.	1 hour
4	**Memory Management**	**(10 hours)**
4.1	Basics - swapping	1 hour
4.2	Memory Allocation (fixed partitions, variable partitions), Fragmentation	1 hour
4.3	Paging	2 hours
4.4	Segmentation	1 hour
4.5	Virtual memory concepts & demand paging	1 hour
4.6	Page replacement algorithms (FIFO, Optimal, LRU).	2 hours
4.7	Allocation of frames, Thrashing	2 hours
5	**Storage Management**	**(6 hours)**
5.1	Introduction , File concept – File Attributes– File Operations, File Types, File structure	1 hours
5.2	File access methods, File allocation methods	1 hour
5.3	Directory structure, Directory implementation	2 hours
5.4	Disk management: Introduction, Disk scheduling algorithms	2 hours

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT305	DATA COMMUNICATION AND NETWORKING	PCC	3	1	0	4

Preamble: The syllabus is prepared with a view to equip the Engineering Graduates to learn basic concepts in data communication and computer networking, and to fine-tune performance parameters used in data transmission.

Prerequisite: Nil

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Discuss the basic concepts used in data communication and computer networking	Level 2 :Understand
CO 2	Identify the concepts of data transmission and apply signal encoding techniques in data transmission.	Level 3 : Apply
CO 3	Compare different transmission mode, multiplexing, and Spread Spectrum techniques.	Level 2 :Understand
CO 4	Describe the design issues and protocols in data link layer.	Level 2 :Understand
CO 5	Summarize the routing algorithms and congestion control techniques in network layer.	Level 2 :Understand

Mapping of Course Outcomes with Program Outcomes 3/2/1: High/Medium/Low

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	1	-	-	-	-	-	-	-	-	2
CO 2	3	3	2	1	2	-	-	-	-	-	-	2
CO 3	2	3	1	2	2	-	-	-	-	-	-	2
CO 4	2	3	3	2	1	-	-	-	-	-	-	2
CO 5	2	2	2	1	1	-	-	-	-	-	-	2

Assessment Pattern

Bloom's Category Levels	Continuous Assessment Tests		End Semester Examination
	1	2	
BL 2: Understand	30	30	60
BL 3: Apply	20	20	40
BL 4: Analyse			
BL 5: Evaluate			
BL 6: Create			

Mark distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be *two* parts; **Part A** and **Part B**. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer *all* questions. Part B contains 2 questions from each module of which student should answer *any one*. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. List the various layers of the OSI reference model.
2. What are the types of topologies used in a network?
3. Mention the various devices used in different layers of the TCP/IP reference model.
4. Define a Protocol Data Unit (PDU).
5. Compare the features of different guided media used in data transmission.
6. Give a comparative analysis of different kinds of satellite communication.
7. Compare and contrast the functionalities of hubs, bridges and switches.

Course Outcome 2 (CO 2):

1. Explain the impairments in data transmission.
2. What is Nyquist criteria for channel bandwidth?
3. Differentiate between analog and digital signals used in transmission.
4. Explain the process of Delta Modulation?

Course Outcome 3 (CO 3):

1. Explain Spread Spectrum Techniques used in networks.
2. Compare and contrast FDM and WDM.
3. Explain CDMA with the help of an example
4. Differentiate statistical TDM and synchronous TDM
5. Discuss synchronous transmission. How is synchronization provided for synchronous transmission?

Course Outcome 4 (CO 4):

1. Assess the suitability of various error correcting codes to deal with single-bit and burst errors in data transmission.
2. Derive a Hamming code for single bit error correction (For a data of length 7 Bit).
3. How are errors detected using parity checking? What are the limitations of parity checking?
4. What are the services offered by the Data Link Layer? Mention the protocols also.
5. With the help of a diagram, explain the format of an Ethernet frame.

Course Outcome 5 (CO 5):

1. What are the functionalities of network layer?
2. Compare distance vector routing and link state routing?
3. What is count-to-infinity problem? How can it be solved?
4. Explain how congestion control is performed in network layer
5. Explain congestion control in virtual circuit subnet

Model Question Paper

Course Code: ITT305

Course Name: Data Communication and Networking

Max.Marks:100 Duration: 3 Hours

Part A

Answer all questions. Each question carries 3 marks. (10 * 3 = 30 Marks)

1. What are the features of WAN.
2. Explain the role of routers in Networks.
3. Explain Data rate, Noise and Bandwidth with respect to a channel.
4. If a periodic signal is decomposed into five sine waves with frequencies of 100, 300, 500, 700, and 900 Hz, what is its bandwidth? Draw the spectrum, assuming all components have a maximum amplitude of 10 V.
5. Draw the constellation diagrams for ASK, BPSK, and QPSK signals.
6. Define scrambling and give its purpose.
7. Using an example, explain two-dimensional parity checks.
8. Write a short note on CDMA.
9. Explain the significance of QoS in communication
10. Explain the importance of the age field in link state messages

Part B

Answer all questions. Each question carries 14 marks. (5 * 14 = 70 Marks)

11		List and explain the main features of all the seven layers of the ISO/OSI reference model and compare it with TCP/IP Model.	14
		OR	
12	a.	Explain the features of any two guided transmission media	9
	b.	Describe the use of satellites in communication	5

13	a.	Explain the features of NRZ, AMI, and Manchester encoding schemes. Encode the given digital data 10110010 using NRZ-L, NRZ-I, AMI, Manchester and differential Manchester encoding schemes?	10
	b.	A telephone line normally has a bandwidth of 3000 Hz (300 to 3300 Hz) assigned for data communications. The signal-to-noise ratio is usually 3162. Find the channel capacity.	4
		OR	
14	a.	What are the transmission impairments happening in data communication?	10
	b.	Consider a channel with a 1-MHz bandwidth. The SNR for this channel is 63. What are the appropriate bit rate and signal level?	4
15	a.	Explain Multiplexing in detail.	10
	b.	List the features of frequency hopping spread spectrum.	4
		OR	
16	a.	Describe direct sequence spread spectrum in detail	10
	b.	Explain in detail about synchronous communication	4
17	a.	List and explain the sliding window protocols used in data link layer	10
	b.	Derive the saturation throughput of pure ALOHA	4
		OR	
18	a.	Describe about CRC encoding and decoding with data word 1010 with $G(x) = x^3+x+1$	10
	b.	What is CSMA/CA?	4
19	a.	Explain distance vector routing in detail	10
	b.	What is flooding?	4
		OR	
20	a	Explain in detail about the congestion control mechanisms used by datagram subnets	10
	b.	What are the services provided by the transport layer?	4

Syllabus

Module 1: Overview of Data Communication and Networks (8 Hours)
Introduction: - Types of Computer Networks, Network Software - Protocol Hierarchies, Connection oriented and Connection less hierarchies, Reference Models - ISO-OSI Reference Model, TCP/IP Reference Model – Comparison of OSI and TCP/IP reference models. Physical Layer: - Guided Transmission Media– Twisted Pair, Coaxial and Fiber Optics, Wireless Transmission- Radio and Microwave transmission, Communication Satellites – GEO, MEO, LEO. Comparison of Network hardware - Repeaters, Routers, Bridges, Gateways, and Hub.
Module 2: Data Transmission and Encoding Techniques (10 Hours)
Data and signals, Analog Signals, Digital Signals - Transmission Impairments,Data Rate Limits: Channel Capacity, Nyquist Bit Rate, Shannon Capacity, Performance parameters - Bandwidth, Throughput, Delay & Jitter. Digital-To-Digital Conversion: Line Coding Schemes: Unipolar, Polar, Bipolar - Block Coding, Scrambling, Analog-To-Digital Conversion: Pulse Code Modulation, Delta Modulation - Digital-To-Analog Conversion: ASK, FSK, PSK.
Module 3: Digital Transmission (7 Hours)
Transmission Modes: Parallel and Serial Transmission, Asynchronous, Synchronous, Isochronous Transmission Multiplexing - TDM, FDM, WDM - Spread spectrum-The concept of spread spectrum – frequency hopping spread spectrum – direct sequence spread spectrum – code division multiple access
Module 4:Link Layer Communication (10 Hours)
Data Link Layer – design issues - Error Detection: Parity Check, Checksum, CRC, Error Correction: Hamming code - Flow Control: Stop-and-Wait, Go-Back-N, and Selective-Repeat - Random Access: ALOHA, CSMA, CSMA/CD, CSMA/CA, Controlled Access: Reservation, Polling, Token Passing, Ethernet- Ethernet Cabling, Encoding, Frame Format, Binary Exponential Back Off Algorithm.
Module 5: Network Layer (10 Hours)
Network Layer Design Issues, Routing Algorithm – Optimality principle - Flooding - Distance vector routing – Link state routing –Multicast Routing - Congestion Control Algorithms – General principles – Congestion prevention policies – Choke packets – Random Early Detection- Quality of Service requirements- Buffering, Traffic shaping – Leaky bucket algorithm. Basic functions of Transport layer and Application layer (Basic understanding only).

Text Books

1. Andrew S. Tanenbaum, Computer Networks, Prentice Hall, 4th Edition, 2003
2. Behrouz A. Forouzan, Data Communications and Networking, 5/e, Tata McGraw Hill, 2017.
3. William Stallings, 'Data and Computer Communications', 8/e Pearson, 2007.

Reference Books

1. William Stallings, Computer Networking with Internet Protocols, Prentice-Hall, 2004.
2. Fred Halsall, Computer Networking and the Internet, 5/e.
3. F. Kurose and K. W. Ross, Computer Networking: A Top-Down Approach Featuring Internet, 6/e, Pearson Education, 2012.
4. L. L. Peterson and B. S. Davie, Computer Networks, A systems approach, 5/e, Morgan Kaufmann, 2011.

Course Contents and Lecture Schedule

Sl. No.	Topic	No. of Lectures
1	**Overview of Data Communication and Networks**	**8 Hours**
1.1	Introduction: - Types of Computer Networks, Network Software - Protocol Hierarchies, Connection oriented and Connection less hierarchies	2
1.2	Reference Models - ISO-OSI Reference Model, TCP/IP Reference Model – Comparison of OSI and TCP/IP reference models	3
1.3	Physical Layer: - Guided Transmission Media– Twisted Pair, Coaxial and Fiber Optics, Wireless Transmission- Radio and Microwave transmission, Communication Satellites – GEO, MEO, LEO. Comparison of Network hardware - Repeaters, Routers, Bridges, Gateways, and Hub.	3
2	**Data Transmission and Encoding Techniques**	**10 Hours**
2.1	Data and signals, Analog Signals, Digital Signals - Transmission Impairments,Data Rate Limits: Channel Capacity, Nyquist Bit Rate, Shannon Capacity, Performance parameters - Bandwidth, Throughput, Delay & Jitter.	4
2.2	Digital-To-Digital Conversion: Line Coding Schemes: Unipolar, Polar, Bipolar - Block Coding, Scrambling,	3
2.3	Analog-To-Digital Conversion: Pulse Code Modulation, Delta Modulation - Digital-To-Analog Conversion: ASK, FSK, PSK.	3
3	**Digital Transmission**	**7 Hours**
3.1	Transmission Modes: Parallel and Serial Transmission, Asynchronous, Synchronous, Isochronous Transmission	2
3.2	Multiplexing - TDM, FDM, WDM	2
3.3	Spread spectrum-The concept of spread spectrum – frequency hopping spread spectrum – direct sequence spread spectrum – code division multiple access	3
4	**Link Layer Communication**	**10 Hours**
4.1	Data Link Layer – design issues	2
4.2	Error Detection: Parity Check, Checksum, CRC, Error Correction: Hamming code	3

4.3	Flow Control: Stop-and-Wait, Go-Back-N, and Selective-Repeat	2
4.4	Random Access: ALOHA, CSMA, CSMA/CD, CSMA/CA, Controlled Access: Reservation, Polling, Token Passing, Ethernet- Ethernet Cabling, Encoding, Frame Format, Binary Exponential Back Off Algorithm.	3
5	**Network Layer**	**10 Hours**
5.1	Network Layer Design Issues, Routing Algorithm – Optimality principle - Flooding	2
5.2	Distance vector routing, Link state routing	2
5.3	Multicast Routing	1
5.4	Congestion Control Algorithms – General principles	1
5.5	Congestion prevention policies – Choke packets – Random Early Detection	2
5.6	Quality of Service requirements- Buffering, Traffic shaping – Leaky bucket algorithm.	1
5.7	Basic functions of Transport layer and Application layer	1

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT307	FORMAL LANGUAGES AND AUTOMATA THEORY	PCC	3	1	0	4

Preamble: The course is considered as a core subject in the area of computer science. This course introduces the formal languages and automata theory which includes various formal languages, strings, finite automaton, grammar, regular expression, pushdown automaton; Linear bounded automata and variants of Turing machine. Not only do they form basic models of computation, they are also the foundation of many branches of computer science, e.g. compilers, NLP, Program verification, Complexity theory. The properties of various automata will be studied and various rigorous techniques for analyzing and comparing them will be discussed, by using both formalism and examples.

Prerequisite: The students are expected to have basic knowledge in Set theory, Functions and Relations.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcomes (CO)	Bloom's Category Level
CO 1	Understand the formal language hierarchy and its applications in the field of computation.	Level 2: Understand
CO 2	Construct automaton for any given regular language and find its equivalent regular expressions.	Level 3: Apply
CO 3	Design a context free grammar for any given context free language.	Level 3: Apply
CO 4	Construct Turing machines and understand their capability.	Level 3: Apply
CO 5	Analyze P,NP class and various undecidable problems.	Level 4: Analyze

Mapping of course outcomes with program outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2										2
CO 2	3	1										3
CO 3	1		3		2							1
CO 4	3	2			1							2
CO 5	3	3										3

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
BL 1: Remember	10	10	20
BL 2: Understand	30	30	60
BL 3: Apply	10	5	15
BL 4: Analyse		5	5
BL 5: Evaluate			
BL 6: Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Define grammar. How can we represent grammar in terms of an expression?
2. Check whether the given grammar is ambiguous or not.
3. Give a brief note on chomsky hierarchy for language classification

Course Outcome 2 (CO2):

1. Construct a DFA for the given regular language.
2. Consider the language for the NFA, strings ending with 'ab' over Σ = {a , b}. Convert the NFA into equivalent DFA.
3. Prove that the following languages are regular or not (Ex: $L = \{a^n \mid n >= 0\}$

Course Outcome 3(CO3):

1. Construct context free grammar for the regular expression:

   L = (a + b)* (aa + bb) (a+b)*

2. Which Normal form representation of CFG will you prefer in converting CFG to NPDA? Why?
3. Can we construct a deterministic PDA for the language ww^R . Justify your answer and also design a NPDA machine for the above language.

Course Outcome 4 (CO4):

1. Write a note on recursively enumerable language.
2. " For every language in the universe, there exists a TM". Justify your answer with a suitable example.
3. Discuss briefly about the halting problem of TM.

Course Outcome 5 (CO5):

1. Discuss on tractable problems.
2. Give a brief note on Universal Turing Machine(UTM).
3. Compare and contrast decidable problems and undecidable problems.

Model Question paper

Course Code: **ITT307**

Course Name: Formal Languages and Automata Theory

Max.Marks:100 Duration: 3 Hours

PART A

(10*3=30)

(Each Question carries 3 Marks)

1. Explain a) Language of DFA b) Extended transition function.
2. Design a DFA, which accepts the strings with even number of 0's and even number of 1's over {0,1}.
3. Construct the finite automaton equivalent to the regular expression i) R.S , ii) R*.
4. Design a Moore machine that takes a set of all strings over {a,b} as input and prints 1 as output for every occurrence of baa as a substring.
5. If a DFA D constructed from NFA N by the subset construction, then L(D) = L(N). Prove it.

6. Design ε -NFA for the set of strings consisting of zero or more a's followed by zero or more b's followed by zero or more c's. Try to use ε transitions to simplify your design.
7. Show that the language L={ 0 i 1 i | i is an integer and i>=1} is not regular using Pumping Lemma.
8. Explain Closure properties of CFL.
9. Define CFG. Give CFG generating the set of palindromes over alphabet {a,b}.
10. Define PushDown Automata.

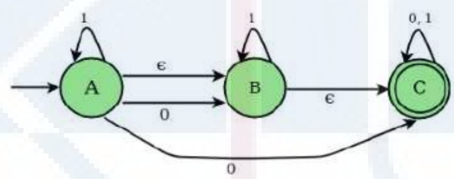
PART B **(5*14=70)**

11. State and prove the equivalence theorem between DFA and NFA.

OR

12. Construct a Moore machine that takes a set of all strings over { a, b } as input and prints '1' as output for every occurrence of ' ab ' as a substring.

13. Convert the following NFA- ∈ to its equivalent DFA.



OR

14. State and prove myhill nerode theorem. Also minimize a dfa for any regular language.

15. Obtain CFG for the language " Set of all palindromes". Discuss it.

OR

16. Can we construct a deterministic PDA for the language ww^R . Justify your answer. Also design a NPDA machine for the above language.

17. Simplify the following grammar:

S->AB / a

A-> BC / b

B-> aC / B

C->aB/ C

OR

18. Discuss about Turing Machine Halting Problem with a suitable example.

19. With a neat sketch explain about Universal Turing Machine.

OR

20. Design a Turing Machine, which can compute the second complement of a given binary number.

Syllabus

Module 1 – Finite automata (9 hours)
Family of formal languages - Finite automata – Type 3 formalism - Deterministic finite automata (DFA) – Language acceptance - Non-deterministic finite automata (NFA) – Finite automata with epsilon transitions – Applications - Finite automata with output - NFA to DFA conversions - Equivalence theorem between DFA and NFA - Minimization of DFA.
Module 2 - Regular languages & Regular expressions (10 hours)
Regular languages and Regular expressions: Myhill-Nerode theorem - Conversion of DFA's to Regular expressions by eliminating states - Conversion of Regular expressions to Automata – Closure properties of Regular languages – Pumping lemma for Regular languages - Applications of the Pumping lemma.
Module 3 – Type 2 formalism & Push Down Automata (10 hours)
Type 2 formalism: Context free grammars (CFG) and languages – Parse trees – Ambiguity in grammars – Pushdown automata (PDA) – Acceptance by final state and empty stack – Equivalence of PDA's and CFG's – Deterministic push down automata (DPDA) – Simplification of CFG - Pumping lemma for CFG's – Chomsky normal form – Greibach normal form.
Module 4 – Type 1 formalism(9 hours)
Closure properties of context free languages – Decision properties of CFL's - Type 1 formalism: Context sensitive grammar – Linear bounded automata .Type 0 formalism: Turing machine (TM) - Recursively enumerable language (REL) – Multitape TM – Non-deterministic TM – Properties of TM.
Module 5 - Undecidability and Universal Turing Machine (7 hours)
Halting problem of TM – Recursive languages - Unrestricted grammars - Universal Turing Machine (UTM) – Tractability - Undecidable problems - Introduction to P and NP class problems.

Text Books

1. J.E.Hopcroft, R.Motwani and J.D.Ullman , "Introduction to Automata Theory Languages and computation",3rd edition Pearson, 2008.
2. Michael Sipser, “Introduction to the Theory of Computation”, 3rd edition (or 1st edition),Course Technology Inc, 2013.

Reference Books

1. Harry R. Lewis, Christos H. Papadimitriou, “ Elements of the Theory of Computation”, Prentice-hall Publisher, 2nd edition, 1998.
2. K.Krithivasan and R.Rama; Introduction to Formal Languages, Automata Theory and Computation ,Pearson Education, 2009.
3. John C . Martin, “Introduction to Languages and the Theory of Computation”, McGraw-Hill Publisher, 4th edition, 2010.
4. Dexter C. Kozen, “Automata and Computability”, Springer.1997.

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Finite automata (9 hours)**	
1.1	Family of formal languages	1
1.2	Deterministic finite automata	2
1.3	Non-deterministic finite automata	2
1.4	Finite automata with epsilon transitions	1
1.5	Finite automata with output	1
1.6	Equivalence between DFA and NFA	1
1.7	Minimization of DFA	1
2	**Regular languages & Regular expressions (10 hours)**	
2.1	Regular languages	1
2.2	Regular expressions	1
2.3	Myhill-Nerode theorem	1
2.4	Conversion of DFA’s to Regular expressions by state elimination	2
2.5	Conversion of Regular expressions to Automata	1
2.6	Closure properties of Regular languages	1
2.7	Pumping lemma for Regular languages	2
2.8	Applications of the Pumping lemma.	1
3	**Type 2 formalism & Push Down Automata (10 hours)**	
3.1	Context free grammars	1
3.2	Ambiguity in grammars	1
3.3	Push down automata(PDA)	2
3.4	String Acceptance by final state and empty stack	1
3.5	Equivalence of PDA’s and CFG’s	1
3.6	Deterministic push down automata (DPDA)	1

3.7	Simplification of CFG	1
3.8	Pumping lemma for CFG's	1
3.9	CNF and GNF	1
4	**Type 1 formalism(9 hours)**	
4.1	Closure properties of context free languages	1
4.2	Context sensitive grammar	1
4.3	Linear bounded automata	1
4.4	Turing machine	2
4.5	Recursively enumerable languages , Properties	1
4.6	Non-deterministic TM	2
4.7	Properties of TM	1
5	**Undecidability and Universal Turing Machine (7 hours)**	
5.1	Halting problem of TM	1
5.2	Recursive languages	1
5.3	Unrestricted grammars	1
5.4	Universal Turing machine	1
5.5	Tractability,Undecidable problems	1
5.6	P, NP class problems	2

****

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT309	MANAGEMENT FOR SOFTWARE ENGINEERS	PCC	3	0	0	3

Preamble: This course aims on providing the concepts of Software Engineering, Software Development Life Cycle and the key aspects of managing a software project like project evaluation, planning, monitoring along with management of people and quality.

Prerequisite: Nil

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category
CO 1	Understand about the basics of software process, software development life cycle and process models.	Level 2 : Understand
CO 2	Interpret the concepts of managing software projects.	Level 2 : Understand
CO 3	Make use of project evaluation techniques and choose software estimation approaches for effort and cost.	Level 3: Apply
CO 4	Explain on planning the project activities and describe the concepts of risk management and resource allocation.	Level 2 : Understand
CO 5	Understand project monitoring and control, organize people and teams and describe the techniques for ensuring software quality.	Level 2 : Understand

Mapping of course outcomes with program outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	2								1	1	2
CO 2	2	2								1	1	2
CO 3	2	1								1	1	2
CO 4	2	1								1	1	2
CO 5	2	2						1	2	1	2	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	10
Understand	20	20	80
Apply	20	20	10
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. What is Software Engineering?
2. Explain about Software Process.
3. Explain about any one process model.

Course Outcome 2 (CO2)

1. Explain any one aspect of software project management spectrum.
2. Explain the importance of software project management over other types of projects.

3. What are the activities involved in project management?

Course Outcome 3(CO3):

1. How evaluation of individual projects is done?
2. What are steps in project planning?
3. Explain any one effort estimation technique.
4. Suppose a project was estimated to be 400 KLOC. Calculate the effort and development time for each of the three model i.e., organic, semi-detached & embedded.
5. Explain on cost estimation.

Course Outcome 4 (CO4):

1. Explain any one scheduling technique.
2. Explain on risk assessment.
3. How are resources identified for a project?

Course Outcome 5 (CO5):

1. How data collection is done for project monitoring?
2. What are the factors for selecting a right person for a project?
3. What is a quality plan?

Model Question paper

Course Code: ITT309

Course Name: MANAGEMENT FOR SOFTWARE ENGINEERS

Max Marks:100 Duration: 3hr

PART A

Answer all questions, each carries 3 marks

1. Explain the Generic process framework for Software engineering.

2. Briefly explain about Scrum.

3. How can a Project manager avoid the problems in a project which may lead to project failure?

4. Compare Traditional and Modern project management practices.

5. What is Cost-benefit analysis?

6. Explain the Bottom-up approach of estimation.

7. What are activity-on-arrow networks?

8. How are Risks identified?

9. What are the activities that are carried out as a part of the project termination review process?

10. Explain CMMI. (10*3=30)

PART B

Answer all questions, each carries 14 marks

11. a) What are the advantages and disadvantages of Waterfall model? How can an Agile model overcome the disadvantages of Waterfall model? (8)

b) What is Agile Modeling? Explain. (6)

OR

12. a) Explain prototyping model in detail. (9)

b) What is Pair programming? What are the advantages of Pair Programming? (5)

13. a) Explain about Agile Teams. (5)

b) How are Software projects categorized? (9)

OR

14. a) Explain the Product aspect of Software management spectrum. (8)

b) Explain the major activities carried out by a software project manager and the order in which these are carried out. (6)

15. Explain in detail the Steps in Project planning. (14)

OR

16. a) What is Benefits management? Explain. (5)

b) Explain the COCOMO II approach for effort estimation. (9)

17. a) Explain on the Forward pass and Backward pass analysis in CPM. (8)

b) How are resources scheduled over the duration of the project? (6)

OR

18. a) What is an Activity? Explain the approaches for identifying the activities that make up a project. (7)

b) Explain any one techniques used for evaluating risks in the project schedule. (7)

19. a) Explain Software Configuration Management. (9)

b) What is leadership? What are the various styles of Leadership? (6)

OR

20. a) Explain in detail about Earned Value Analysis. (10)

b) Explain the difference between Verification and Validation. (4)

Syllabus

Module 1 (8 Hours)
Introduction: Software engineering, Software process, Software engineering practice **Process models**: Prescriptive process models- Specialised process models, The unified process, Personal and Team process models. **Agile development:** Agility, Agile process. Extreme programming- XP Values, The XP Process, Industrial XP, The XP Debate. Other Agile development models- Adaptive Software Development (ASD), Scrum, Dynamic Systems Development Method (DSDM), Crystal, Feature Driven Development (FDD), Lean Software Development (LSD), Agile Modeling (AM), Agile Unified Process (AUP). Selection of an appropriate Project Approach- Choice of process Models.
Module 2 (5 Hours)
Managing software projects: Project Management Concepts –The Management Spectrum- People- Product- Process- Project. **Software Project Management** - Importance – Software projects VS other types of project – Categorizing Software projects- Stakeholders – Setting Objectives –The Business Case- Project success and failure. **Management** –Activities- Management Control- Traditional VS modern project management.
Module 3 (8 Hours)
Project Evaluation: Project portfolio management- Evaluation of individual projects- Cost benefit evaluation techniques- Risk evaluation- Programme Management- Creating a Programme- Aids to Programme Management- Benefits Management. **Project Planning**: Step wise Project Planning **Software Estimation:** Basis for software estimation- Software Effort estimation techniques- Bottom-up and Top-down estimation- Function Point Analysis- COCOMO II. Cost Estimation- Staffing Pattern- Schedule compression.

Module 4 (7 Hours)
Activity Planning: Objectives- Project Schedules- Projects and Activities- Sequencing and Scheduling Activities- Network Planning Models- Forward Pass- Backward pass- Identifying Critical Path and Critical Activities- Activity-on-arrow networks. **Risk Management:** Risk- Categories of Risk- Risk Identification- Risk Assessment- Risk Planning- Risk management- Risk Evaluation- PERT, Monte Carlo Simulation, Critical Chain. **Resource Allocation:** Nature of Resources- Identifying and Scheduling Resources- Creating Critical Paths- Cost Schedule- Scheduling sequence.
Module 5 (7 Hours)
Monitoring and Control: Creating the framework- Collecting data- Review- Project Termination Review- Visualizing Progress- Gantt Chart, Slip Chart, Timeline. Cost Monitoring- Earned Value Analysis- Getting the project back to target- Change control- Software Configuration Management- Contract management. **Managing People:** Organizational Behaviour- Selecting the right Person- Motivation- Stress- Working in Teams- Becoming a Team- Decision Making- Organization and Team Structures- Communication- Leadership. **Software Quality:** Quality Management Systems- Process Capability Models- CMMI, Six Sigma. Techniques for Enhancing Software Quality- Testing- Software Reliability- Quality Plans.

Text Books

1. Roger S Pressman, Software Engineering: A Practitioner's Approach, Seventh edition, Tata McGraw Hill.
2. Bob Hughes, Mike Cotterell and Rajib Mall: Software Project Management – Fifth Edition, Tata McGraw Hill

Reference Books

1. Pankaj Jalote, Software Project Management in Practice, Pearson Education
2. Walker Royce, Software Project Management- Addison-Wesley, 1998.
3. Sunitha E.V, Sarath K.S, Software Project Management, Jyothis Publishers 2019

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction**	**8 Hours**
1.1	Software engineering, Software process, Software engineering practice	1
1.2	**Process models**: Prescriptive process models- Specialised process models, The unified process, Personal and Team process models.	3
1.3	**Agile development:** Agility, Agile process. Extreme programming- XP Values, The XP Process, Industrial XP, The XP Debate. Other Agile development models- Adaptive Software Development (ASD), Scrum, Dynamic Systems Development Method (DSDM), Crystal, Feature Driven Development (FDD), Lean Software Development (LSD), Agile Modeling (AM), Agile Unified Process (AUP). Selection of an appropriate Project Approach- Choice of process Models.	4
2	**Managing software projects**	**5 Hours**
2.1	Project Management Concepts –The Management Spectrum- People- Product- Process- Project.	1
2.2	**Software Project Management** - Importance – Software projects VS other types of project – Categorizing Software projects- Stakeholders – Setting Objectives –The Business Case- Project success and failure.	3
2.3	**Management** –Activities- Management Control- Traditional VS modern project management.	1
3	**Evaluation, Planning and Estimation**	**8 Hours**
3.1	**Project Evaluation:** Project portfolio management- Evaluation of individual projects- Cost benefit evaluation techniques- Risk evaluation- Programme Management- Creating a Programme- Aids to Programme Management- Benefits Management.	3
3.2	**Project Planning**: Step wise Project Planning	2
3.3	**Software Estimation:** Basis for software estimation- Software Effort estimation techniques- Bottom-up and Top-down estimation- Function Point Analysis- COCOMO II. Cost Estimation- Staffing Pattern- Schedule compression.	3
4	**Activity Planning, Risk management and Resource allocation**	**7 Hours**
4.1	**Activity Planning:** Objectives- Project Schedules- Projects and Activities- Sequencing and Scheduling Activities- Network	3

	Planning Models- Forward Pass- Backward pass- Identifying Critical Path and Critical Activities- Activity-on-arrow networks.	
4.2	**Risk Management:** Risk- Categories of Risk- Risk Identification- Risk Assessment- Risk Planning- Risk management- Risk Evaluation- PERT, Monte Carlo Simulation, Critical Chain.	2
4.3	**Resource Allocation:** Nature of Resources- Identifying and Scheduling Resources- Creating Critical Paths- Cost Schedule- Scheduling sequence	2
5	**Monitoring, People management, Quality**	**7 Hours**
5.1	**Monitoring and Control:** Creating the framework- Collecting data- Review- Project Termination Review- Visualizing Progress- Gantt Chart, Slip Chart, Timeline. Cost Monitoring- Earned Value Analysis- Getting the project back to target- Change control- Software Configuration Management- Contract management.	3
5.2	**Managing People:** Organizational Behaviour- Selecting the right Person- Motivation- Stress- Working in Teams- Becoming a Team- Decision Making- Organization and Team Structures- Communication- Leadership.	2
5.3	**Software Quality:** Quality Management Systems- Process Capability Models- CMMI, Six Sigma. Techniques for Enhancing Software Quality- Testing- Software Reliability- Quality Plans.	2

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITL331	OPERATING SYSTEM AND NETWORK PROGRAMMING LAB	PCC	0	0	3	2

Preamble: Operating System and Network Programming Lab aims at giving an in depth idea of operating system and networking concepts. Students will understand the basic commands and the implementation of process scheduling, inter process communication, semaphores etc. and also aim to implement network programming in Java.

Prerequisite: Concepts of Operating Systems and Networking, and Programming knowledge in C and JAVA

Course Outcomes:

After the completion of the course the student will be able to

CO. No.	Course Outcomes	Bloom's Taxonomy
CO 1	Analyse CPU Scheduling Algorithms like FCFS, Round Robin, SJF and Priority.	Level 4: Analyse
CO 2	Implement inter process communication and process synchronization problems.	Level 3: Apply
CO 3	Implement memory management schemes - first fit, best fit and worst fit.	Level 3: Apply
CO 4	Implement client server communication using sockets.	Level 3: Apply
CO 5	Implement MAC protocols.	Level 3: Apply
CO 6	Familiarization of network simulation tool.	Level 2: Understand

Mapping of course outcomes with program outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3	3	3	2	-	-	-	-	1	-	3
CO 2	3	3	3	2	1	-	-	-	-	1	-	3
CO 3	3	3	3	2	1	-	-	-	-	1	-	3
CO 4	3	3	3	2	2	-	-	-	-	1	-	3
CO 5	3	3	3	2	2	-	-	-	-	1	-	3
CO 6	2	2	2	2	3	-	-	-	-	1	-	3

3/2/1: High/Medium/Low**6**

Assessment Pattern:

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	75	75	2.5 hours

Continuous Internal Evaluation Pattern:

Attendance : 15 marks
Continuous Assessment : 30 marks
Internal Test (Immediately before the second series test) : 30 marks

End Semester Examination Pattern: The following guidelines should be followed regarding award of marks

(a) Preliminary work : 15 Marks
(b) Implementing the work/Conducting the experiment : 10 Marks
(c) Performance, result and inference (usage of equipments and trouble shooting) : 25 Marks
(d) Viva voce : 20 marks
(e) Record : 5 Marks

General instructions: Practical examination to be conducted immediately after the second series test covering entire syllabus given below. Evaluation is a serious process that is to be conducted under the equal responsibility of both the internal and external examiners. The number of candidates evaluated per day should not exceed 20. Students shall be allowed for the University examination only on submitting the duly certified record. The external examiner shall endorse the record.

Course Level Assessment Questions:

Course Outcome 1 (CO1):

1. Write a program to implement FCFS and Round Robin process scheduling algorithms for the following scenario.

Processes	Arrival time	Burst time
P0	0	7
P1	2	4
P2	4	1
P3	5	4

Assume quantum time for RR is 2

(i) What is the Average Turn-around time for each of these scheduling algorithms?

(ii) What is the Total Waiting time for each of these scheduling algorithms?

Course Outcome 2 (CO2):
1. Program to implement Inter Process Communication using shared memory.
2. Program to implement Dining Philosophers problem using semaphores.

Course Outcome 3 (CO3):
1. Implement first fit, best fit and worst fit memory management schemes.

Course Outcome 4 (CO4):
1. Program to implement client server communication using sockets.
2. Program to implement chat application.

Course Outcome 5 (CO5):
1. Program to implement Go Back N protocol.

Course Outcome 6 (CO6):
1. Simulate Bus topology using NS-3.

LIST OF EXPERIMENTS

(All the listed experiments are mandatory)

OPERATING SYSTEM

(Experiments are to be implemented using C programming language)

1. Familiarization of system calls (fork, exec, getpid, exit, wait, close, stat etc) in operating system.
2. Implement process scheduling algorithms (FCFS, SJF, Round-Robin, Priority) and compute average waiting time and average turn-around time.
3. Inter-process communication using mail boxes, pipes, message queues and shared memory.
4. Implementation of dining philosophers problem using threads, semaphores and shared memory.
5. Implementation of banker's algorithm.
6. Implement memory management schemes (first fit, best fit and worst fit).

NETWORK PROGRAMMING

(Experiments are to be implemented using JAVA programming language)

7. Familiarisation of Network Programming API in Java.
8. Implementation of Medium Access Control protocols – 1) Go Back N. 2) Selective Repeat and 3) Sliding Window.
9. Implementation of an echo server.
10. Implement Client-Server communication using sockets.
11. Implementation of chat application
12. Install network simulator NS-3 in Linux operating system and simulate wired and wireless scenarios. (Familiarization only)

References:

1. Abraham Silberschatz, Peter B Galvin, Greg Gagne, Operating System Concepts, 9/e, Wiley India, 2015.
2. Behrouz A Forouzan, Data Communications & Networking –Mc Graw Hill,2008.
3. Herbert Schildt, "The Java 2 : Complete Reference" Tenth Edition - Mc Graw Hill.
4. https://www.nsnam.org/docs/tutorial/html/

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITL333	WEB APPLICATION DEVELOPMENT LAB	PCC	0	0	3	2

Preamble: Web Application Development Lab is intended to deliver hands-on experience of Web Application Development with HTML, CSS, JavaScript, JQuery, Node JS and Mongo DB thereby equipping them to develop real time web applications.
Prerequisites: Basics of Programming, ITT301 Web Application Development

Course Outcomes: After the completion of the course the student will be able to

CO.No.	Course Outcomes
CO1	Infer the structure of HTML elements in a webpage
CO2	Build Webpages using HTML and CSS
CO3	Utilize JavaScript to add functionality to webpages
CO4	Implement different Ajax & JQuery functionalities in Web development.
CO5	Develop a web applications using Node JS and MongoDB

Mapping of Course Outcomes with Program Outcomes

	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO 8	PO9	PO1 0	PO1 1	PO1 2
CO1	1	2	-	2	-	-	-	-	-	-	-	-
CO2	1	2	-	2	-	-	-	-	-	-	-	-
CO3	2	2	-	2	-	-	-	-	-	-	-	-
CO4	2	2	-	3	-	-	-	-	-	-	-	-
CO5	2	3	1	3	-	-	-	-	-	-	-	2

3/2/1: High/Medium/Low

Assessment Pattern

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	75	75	2.5 hours

Continuous Internal Evaluation Pattern:

Attendance	:	15 marks
Continuous Assessment	:	30 marks
Internal Test (Immediately before the second series test)	:	30 marks

End Semester Examination Pattern:

The following guidelines should be followed regarding award of marks

(a) Preliminary work	: 15 Marks
(b) Implementing the work/Conducting the experiment	: 10 Marks
(c) Performance, result and inference (usage of equipment and troubleshooting)	: 25 Marks
(d) Viva voce	: 20 Marks
(e) Record	: 5 Marks

General instructions: Practical examination to be conducted immediately after the second series test covering entire syllabus given below. Evaluation is a serious process that is to be conducted under the equal responsibility of both the internal and external examiners. The number of candidates evaluated per day should not exceed 20. Students shall be allowed for the University examination only on submitting the duly certified record. The external examiner shall endorse the record.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Develop a website (HTML) for College Library having pages:
    a) Home Page:
        - Navbar
        - Banner image
        - Contents
        - Footer
    b) Books Management
        - Add/Remove Book
        - Book Details
    c) About Us

2. Create a website which demonstrates the usage of following HTML tags:
    - Headings
    - Paragraphs
    - Image
    - Lists
    - Anchor
    - Forms
3. Develop a website for Online Bus Ticket Booking having pages:
    a) Home Page:
        - Navbar

-Banner image
- Contents
- Footer

b) Ticket Booking
- Book Ticket
- Booking Details

c) About Us

Course Outcome 2 (CO2)

1. Develop a website for College Library having pages:

a) Home Page:
- Navbar
- Banner image
- Contents
- Footer

b) Books Management
- Add/Remove Book
- Book Details

c) About Us

2. Develop a Responsive website for Online Bus Ticket Booking

3. Develop an Online shopping website using HTML and Bootstrap

Course Outcome 3(CO3):

1. Write javascript code to calculate grades of students and average grade of a class(use prompt to get input)

2. Develop "Craps dice game" with javascript.

3. Develop a to do list app with HTML, CSS and Javascript (use AJAX with JSON)

Course Outcome 4 (CO4):

1. Develop Craps dice game with jquery

2. Develop a to do list app with HTML, CSS and JQuery(use AJAX with JSON)

3. Develop an online shopping website with HTML, CSS and JQuery (use JQuery Animations and plugins)

Course Outcome 5 (CO5):

1. Develop a website for College Library using Node JS with MongoDB.

2. Develop a Responsive website for Online Bus Ticket Booking using Node JS with MongoDB

3. Develop an Online shopping website using Node JS with MongoDB

LIST OF EXPERIMENTS

(All the listed experiments are mandatory)

1. Install, setup Integrated Development Environment (IDE) for web development.
2. Create a web page with all possible elements of HTML5
3. Create a web page with all types of Cascading style sheets
4. Create a Responsive Web page with HTML and CSS
5. Create Responsive web page with Bootstrap
6. Programs to demonstrate JavaScript array, object and functions
7. Client Side Scripts for Form Validation using JavaScript
8. Programs to familiarise ES6 concepts
9. Programs to demonstrate DOM and event handling.
10. Programs using AJAX with HTML, XML and JSON data
11. Programs to familiarise JQuery.
12. Create a website with HTML, CSS and Javascript (implement Ajax)
13. Programs to familiarise Server Side Scripting using Node JS
14. Programs using MongoDB database with Node JS
15. Develop a web site with HTML, CSS, Javascript/JQuery, Node JS and MongoDB

Reference Books

1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, “Internet and World Wide Web How To Program”, 5/E, Pearson Education, 2012.
2. Jon Duckett , “HTML and CSS: Design and Build Websites”, Wiley
3. Jon Duckett , “JavaScript and JQuery: Interactive Front–End Web Development”, Wiley
4. Nicholas C. Zakas, “Understanding ECMAScript 6: The Definitive Guide for JavaScript Developers”
5. Alex Young, Marc Harter, “Node js in practice”, Manning
6. Json Krol , “Web Development with mongodb and node js”, Packt
7. Krishna Rungta , “Node JS: learn in one day”

SEMESTER V

MINOR

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT381	WEB APPLICATION DEVELOPMENT	VAC	3	1	0	4

Preamble: This course is intended to make the students capable of understanding the important components of HTML5 documents and use HTML5,CSS to create web pages.The course is intended to help he students to develop dynamic web pages PHP, Java Servlets and Java Server Pages

Prerequisite: ITT281 JAVA programming, ITT282 Database Management

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcomes	Bloom's Category
CO1	Discuss the important components of HTML5 documents and use HTML5 to create web pages	Level 2: Understand
CO2	Apply styles in web pages using cascading style sheets	Level 3: Apply
CO3	Develop dynamic web pages using PHP.	Level 3: Apply
CO4	Develop server based programs using Java Servlets	Level 3: Apply
CO5	Develop dynamic web pages using Java Server Pages	Level 3: Apply

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	-	3	-	2	-	-	-	-	-	-	3
CO 2	2	-	3	-	2	-	-	-	-	-	-	3
CO 3	3	-	3	-	2	-	-	-	-	-	-	3
CO 4	3	-	3	-	2	-	-	-	-	-	-	3
CO 5	3	-	3	-	2	-	-	-	-	-	-	3

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	5	5	10
Understand	20	20	30

Apply	25	25	60
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Create an HTML5 document containing an ordered list of three items—ice cream, pizza and soft drinks. Each ordered list should contain a nested, unordered list of your favourite flavours. Provide three flavours in each unordered list.

2. What do you meant by MIME? Explain its uses? Provide any FIVE examples.

3. Create a web page using HTML5 to accomplish a feedback on a cafeteria.

Course Outcome 2 (CO2):

1. Using CSS create a sophisticated drop down menu for a set of web page addresses

2. Explain conflicting styles with examples. What happens when conflicting occurs? How can you remove the conflicts?

3. Write a CSS rule that changes the colour of all elements containing attribute class = "red_colour" to red.

.

Course Outcome 3(CO3):

1. With a PHP code snippet illustrate the use of "foreach" loop in PHP.
2. Design and Develop a mail registration form using PHP.
3. Develop a student management system using PHP.

Course Outcome 4(CO4):

1. Design and Develop a Servlet based web application to update the basic salary of all employees belonging to the department of sales by 5%, assuming there exists an employee table with field(e_id,e_dept,e_name,b_sal,n_sal)?
2. Explain the methods used to implement the life cycle of Servlets
3. List out the benefits of Servlets

Course Outcome 5(CO5):

1. How to pass control from one JSP page to another?
2. Explain the importance of data sharing among JSP Pages. Design and Develop a JSP based web application to display the values which is being entered by the user in a registration form
3. How does Error handling is done in JSP

Model Question paper

Course Code: ITT381

Course Name: WEB APPLICATION DEVELOPMENT

Max.Marks:100 **Duration: 3 Hours**

PART A

Answer all Questions. Each question carries 3 Marks

1. What is Internal Linking? How can it be achieved?
2. List out any 5 page structure elements
3. List any 4 media types available in CSS?
4. How can you insert CSS codes in your HTML page? Provide one example for each methods
5. In PHP how can you search a string using regular expressions
6. Discuss the differences between server side programming and client side programming
7. Compare and Contrast doGet and doPost service methods of Servlets

8. Explain the use of Servlets in MVC architecture
9. How to pass control from one JSP page to another?
10. Explain the role of JSP in MVC design

Part B

Answer any one Question from each module. Each question carries 14 Marks

11. a) Suppose your HTML page contains a text input element for inputting months of a year. How can you provide a drop-down list of pre-defined options of months for that text element? (7Marks)

b) Using HTML5 scripting create the below given table (CO1)

Sl. No	Department	No of Students	
		Boys	Girls
1	IT	110	135
2	ME	220	18
3	EC	160	180
4	CE	200	120

(7 Marks)

12. a) Create an HTML5 document containing an ordered list of animals of three kinds—Carnivorous, Herbivorous and Omnivorous. Each ordered list should contain a nested, unordered list of your favourite animals. Provide atleast three of them in each unordered list (8 Marks)

b) Provide HTML tags for inserting the following:

i) >
ii) ©
iii) ®
iv) 1/4
v) Horizontal Rule
vi) &. (6 Marks)

13. a) Illustrate different flavours of positioning elements available in CSS with examples. (7 Marks)

b) Provide CSS rules to set background image of a page. Make it tiled.

(7 Marks)

14.a) Explain conflicting styles with examples. What happens when conflicting occurs? How can you remove the conflicts?
(8 marks)

b) Which all are the media types available in CSS? Explain the uses of atleast two types with an example. (6 Marks)

15. a) Implement a database based online student management system using PHP. The system should have the following features:

i. Provision to input the student details such as stud_name, stud_rollno, stud_age, stud_branch, stud_gender
ii. Provision to search a student using stud_rollno
iii. Provision to delete a student using stud_rollno
iv. Provision to display the details of all students available in the database.

(10 marks)

b) List out the methods to access a web server (4 Marks)

16. a) How can you make data type conversions in PHP? (4 marks)

b) Describe the steps involved in PHP to access a database. (10 Marks)

17. a) Describe the use of Servlet Container with a neat diagram (6 marks)

b) Develop a Servlets based online student management system with the following features:

(i) Student registration (ii) Student search using stud_ID

(iii) Student deletion using stud_ID (8 Marks)

18. a) With a neat diagram explain the Servlets life cycle. (7 marks)

b) Implement a simple mail registration application using Java Servlets. (7 Marks)

19. a) With a JSP program explain the method of sharing control among different pages.

(8 Marks)

b) Develop a registration and login form for an e-mail application using JSP and required database. (6 marks)

20. a) With neat diagrams explain the architecture of JSP applications. (8 Marks)

b) How does error handling done in JSP using Exception objects? Explain with an example. (6 marks)

Syllabus

Module 1	No. of Lectures
Introduction to Computers and the Internet- Web Basics, Introduction to HTML5 - W3C HTML5 Validation Service, Headings, Linking, Images, Special Characters and Horizontal Rules, Lists, Tables, Forms, Internal Linking, meta elements, New HTML5 Form input Types, input and data list elements and auto complete Attribute, Page-Structure Elements.	8 hours
Module 2	
Introduction to Cascading Style Sheets -Inline Styles, Embedded Style Sheets, Conflicting Styles, Linking External Style Sheets, Positioning Elements - Absolute Positioning, z-index, Relative Positioning, span, Backgrounds, Element Dimensions, Box Model and Text Flow, Media Types, Drop-Down Menus	8 hours
Module 3	
Web Servers: Introduction, HTTP Transactions, Multitier Application Architecture, Client-Side Scripting versus Server-Side Scripting, Accessing Web Servers. Server Side Programming with PHP - Introduction, converting Between Data Types, Arithmetic Operators, Form Processing and Business Logic, Using PHP to Process HTML5 Forms, Accessing MySQL Database with PHP	10 hours
Module 4	
Servlets: Introduction to Servlets, Benefits of Servlets, servlets as controller in MVC, basic HTTP, servlet container, servlet lifecycle,Servlets with JDBC	9 hours
Module 5	
Java Server Pages: Generating Dynamic Content, Using Scripting Elements, Implicit JSP Objects. Error Handling and Debugging, Passing Control and Data between Pages – Sharing Session and Application Data – Application Models - MVC Design	10 hours

Text Books

1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, “Internet and World Wide Web How To Program”, 5/E, Pearson Education, 2012.

2. Hans Bergsten , Java Server Pages, O’Reilly, 2003

3. Jason Hunter, William Crawford, Java Servlet Programming, Second Edition, , O'Reilly Media.

Reference Books

1. Robert W. Sebesta, "Programming the World Wide Web", 8/E, Pearson Education, 2012.

2. Chris Bates, "Web Programming – Building Intranet applications", Wiley Publications, 3rd Edition, 2009..

3. Joseph J. Bambara, Paul R. Allen, Mark Ashnault, Ziyad Dean, Thomas Garben, Sherry Smith J2EE UNLEASHED — SAMS Techmedia

4. Roman, Scott Ambler, Tyler Jewell (ed.), Mastering EJB(2nd Edition) – Ed– John Wiley Publications, 2003.

5. Stepahnie Bodoff, Dale Green, Kim Hasse, Eric Jendrock, Monica Pawlan, Beth Stearns , The J2EE Tutorial, Pearson Education , Asia.

6. www.w3schools.com

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction to Computers and the Internet**	**8 hours**
1.1	Introduction to Computers and the Internet- Web Basics	1 hour
1.2	Introduction to HTML5 - W3C HTML5 Validation Service, Headings, Linking, Images, Special Characters and Horizontal Rules	2 hours
1.3	Lists, Tables	1 hour
1.4	Forms, Internal Linking, meta elements	1 hour
1.5	New HTML5 Form input Types, input and data list elements and auto complete Attribute,	2 hours
1.6	Page-Structure Elements.	1 hour
2	**Introduction to Cascading Style Sheets**	**8 hours**
2.1	Inline Styles, Embedded Style Sheets	2 hours
2.2	Conflicting Styles, Linking External Style Sheets	2 hours
2.3	Positioning Elements - Absolute Positioning, z-index, Relative Positioning, span, Backgrounds, Element Dimensions	2 hours
2.4	Box Model and Text Flow, Media Types, Drop-Down Menus	2 hours
3	**Web Servers**	**10 hours**
3.1	Introduction, HTTP Transactions, Multitier Application Architecture, Client-Side Scripting versus Server-Side Scripting, Accessing Web Servers.	2 hours
3.2	Server Side Programming with PHP - Introduction, converting	2 hours

	Between Data Types, Arithmetic Operators	
3.3	Form Processing and Business Logic	2 hours
3.4	Using PHP to Process HTML5 Forms	2 hours
3.5	Accessing MySQL Database with PHP	2 hours
4	**Java Servlets**	**9 hours**
4.1	Introduction to Java Servlets, Benefits of Servlets, use as controller in MVC,	2 hours
4.2	basic HTTP, servlet container,	2 hours
4.3	Servlet lifecycle	1 hour
4.4	Servlets with JDBC	4 hours
5	**Java Server Pages**	**10 hours**
5.1	Generating Dynamic Content, Using Scripting Elements,	2 hours
5.2	Implicit JSP Objects.	1 hour
5.3	Error Handling and Debugging	2 hours
5.4	Passing Control and Data between Pages	2 hours
5.5	Sharing Session and Application Data	2 hours
5.6	Application Models - MVC Design	1 hour

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT383	INTERNET TECHNOLOGY	VAC	3	1	0	4

Preamble: This subject provides an overview on the principles on which the Internet and other distributed systems are based; their architecture, algorithms and design. It covers the important topic of middleware, examining different approaches to supporting distributed applications including distributed objects and components, and web services.The two dominant modern network architectures are cloud computing and the Internet of things (IoT) is also introduced here. The subject then covers the well-established topics of security.

Prerequisite: ITT 283 Data Communication and ITT 284 Computer Networks

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcomes(CO)	Bloom's Category
CO 1	Describe building blocks of distributed systems.	Level2: Understand
CO 2	Explain the functions of remote invocation, operating system and webserver and understand its application on Internet.	Level2:: Understand
CO 3	Familiarize with the basic concepts, cloud services, deployment models, and architecture of cloud computing.	Level2:: Understand
CO 4	Discuss the key foundation and uses of IoT enabled devices and familiarize with the IoT architecture reference model.	Level2:: Understand
CO 5	Describe the modern networking security issues, and their solutions.	Level2:: Understand

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	-	-	-	-	-	-	-	-	-	2
CO 2	3	2	1	-	-	-	-	-	-	-	-	2
CO 3	3	2	1	-	1	-	-	-	-	-	-	2
CO 4	3	2	1	-	-	-	-	-	-	-	-	2
CO 5	3	2	1	-	2	-	-	-	-	-	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test 1(Marks)	Test 2(Marks)	
Remember	10	10	20
Understand	40	40	80
Apply			
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. What are the features of distributed systems?
2. What is interprocess communication?

Course Outcome 2 (CO2)

1. What is request reply protocol?
2. What are the differences between process and threads?

Course Outcome 3(CO3):

1. What are the features of cloud computing?
2. Explain ITU-T Cloud Computing Functional Reference Architecture

Course Outcome 4 (CO4):

1. Explain in detail about types of sensors
2. Explain the elements of RFID systems

Course Outcome 5 (CO5):

1. Discuss the importance of encryption in communication
2. Explain wireless network security

Model Question Paper

Course Code: ITT383

Course Name: INTERNET TECHNOLOGY

Max. Marks: 100 Duration: 3 Hours

PART A

Answer all questions. Each question carries 3 marks. (10 * 3 = 30 Marks)

1. Give examples for distributed system
2. What is HTML?
3. What is middleware?
4. What is a thread?
5. What are the characteristics of cloud computing?
6. What is Infrastructure as a service?
7. What is an actuator?
8. What is RFID?
9. What is the difference between HTTP and HTTPS?
10. What is VPN?

Part B

Answer all questions. Each question carries 14 marks. (5 * 14 = 70 Marks)

11. What is a distributed system? Explain the challenges in distributed system.

OR

12. Explain in detail about overlay networks.

13. Explain in detail about remote procedure call.

OR

14. Explain about web service infrastructure and components.

15. Illustrate NIST cloud computing reference architecture.

OR

16. Describe the three basic cloud services in detail.

17. Explain the different elements in an RFID system in detail.

OR

18. Describe the ITU-T Y.2060 IoT Reference Model.

19. Explain any 5 security scams used to fool users

OR

20. Explain in detail about wireless network security

Syllabus

Module 1: Introduction to Distributed Systems (9Hours)
Characterization of Distributed Systems: Introduction, Examples of distributed systems, Trends in distributed systems, Challenges, Case Study: The World Wide Web. System Models: Introduction, Physical Models, Architectural models. Interprocess Communication: Introduction, The APIs for internet protocols, External data representation and marshalling, Network Virtualization: Overlay networks. *Text Book I – Chapter 1,2, and 4*
Module 2: Distributed Middleware Application(9 Hours)
Remote Invocation: Introduction, request-reply protocols, remote procedure call, remote method invocation. Operating system support: Introduction, The operating system layer, Processes and threads, Communication and invocation. Web Services: Introduction, Web Services, Coordination of web services, Application of web services *Text Book I – 5, 7, and 9*
Module 3: Cloud Computing (9 Hours)
Basic Concepts, Cloud Computing Elements, Cloud Service Models: SaaS, PaaS, IaaS, Other Cloud Services, Cloud Deployment Models, NIST Cloud Computing Reference Architecture, ITU-T Cloud Computing Reference Architecture, ITU-T Cloud Computing Functional Reference Architecture *Text Book II – Chapter 13*

Module 4: Internet of Things (9Hours)
The Internet of Things: Components. Scope of the Internet of Things, Components of IoT-Enabled Things: Sensors, Types of Sensors, Actuators, Embedded System, Microprocessors, Microcontrollers, Transceivers, RFID, IoT Architecture - ITU-T Y.2060 IoT Reference Model, IoT World Forum Reference Model *Text Book II – Chapter 14, 15*
Module 5: Network Security(9 Hours)
Introduction, Cybercrime And Cyber Security, Unsecure Internet, Computer Encryption, Confidential Web Browsing, Encryption Keys, Authentication: User IDs And Passwords, Two-Factor Authentication,Wireless Network Security, Network Firewall, Security Scams, Man-In-The-Middle Attacks, Email Addresses And Web Site URLs based attacks, Malware In Email Attachments, Secure Access with VPNs, VPN Technology *Text Book III – Chapter 14, 15*

Text Books

1. George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair, “Distributed Systems – Concepts and Design”. 5/e Addison Wesley, Inc., 2012.
2. William Stallings, “Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud”. 10/e Pearson Education, Inc., 2016.
3. Douglas E. Comer, “The Internet Book: Everything You Need to Know about Computer Networking and How the Internet Works”, 5th edition, CRC Press, 2019.

Reference Books

1. William Stallings, “Computer Security: Principles and Practice”, 3/e, Pearson Education Inc,2015
2. Daniel Minoli, “Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications”, Wiley, 2013
3. Toby Velte, Anthony Velte, Robert Elsenpeter, “Cloud Computing, A Practical Approach”, McGrawhill, 2009
4. Douglas E. Comer, Computer Networks and Internets, 6/e,Pearson Education Inc,2018

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Module 1: Introduction to Distributed Systems (9 Hours)**	
1.1	Characterization of Distributed Systems: Introduction, Examples of distributed systems	2Hours
1.2	Trends in distributed systems, Challenges, Case Study: The World Wide Web.	2Hours
1.3	System Models: Introduction, Physical Models, Architectural models.	2Hours
1.4	Interprocess Communication: Introduction, The APIs for internet protocols, External data representation and marshalling	2 Hours
1.5	Network Virtualization: Overlay networks	1 Hour
2	**Module 2: Distributed Middleware Application(9 Hours)**	
2.1	Remote Invocation: Introduction, request-reply protocols	2 Hours
2.2	Remote procedure call, remote method invocation.	2 Hours
2.3	Operating system support: Introduction, The operating system layer, Processes and threads, Communication and invocation.	3 Hours
2.4	Web Services: Introduction, Web Services	1 Hours
2.5	Coordination of web services, Application of web services	1 Hours
3	**Module 3: Cloud Computing (9 Hours)**	
3.1	Basic Concepts, Cloud Computing Elements, Cloud Service Models: SaaS, PaaS, IaaS, Other Cloud Services	3Hours
3.2	Cloud Deployment Models	2 Hours
3.3	NIST Cloud Computing Reference Architecture, ITU-T Cloud Computing Reference Architecture, ITU-T Cloud Computing Functional Reference Architecture	4Hours
4	**Module 4: Internet of Things (9Hours)**	
4.1	The Internet of Things: Components. Scope of the Internet of Things	3Hours
4.2	Components of IoT-Enabled Things: Sensors, Types of Sensors, Actuators, Embedded System, Microprocessors, Microcontrollers, Transceivers, RFID	3 Hours
4.3	IoT Architecture - ITU-T Y.2060 IoT Reference Model, IoT World Forum Reference Model	3 Hours
5	**Module 5: Security (9Hours)**	
5.1	Introduction, Cybercrime And Cyber Security	1Hour
5.2	Unsecure Internet, Computer Encryption, Confidential Web Browsing, Encryption Keys	1Hour
5.3	Authentication: User IDs And Passwords, Two-Factor Authentication	2Hours
5.4	Wireless Network Security, Network Firewall	2Hours

5.5	Security Scams, Man-In-The-Middle Attacks, Email Addresses And Web Site URLs based attacks, Malware In Email Attachments	2 Hours
5.6	Secure Access with VPNs, VPN Technology	1 Hour

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT385	SOFTWARE ARCHITECTURE CONCEPTS	VAC	3	1	0	4

Preamble: The syllabus is prepared with the view of preparing the Engineering Graduates capable ofunderstanding essential concept of software architecture.

Prerequisite: Basic programming knowledge

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category
CO1	Summarize role of software architecture and architect	Level 2: Understand
CO2	Discuss Basic Concepts in Software Architecture	Level 2: Understand
CO3	Explain Design Patterns	Level 2: Understand
CO4	Explain the role of Architecture in SDLC	Level 2: Understand
CO5	Identify the role of Architecture in Business	Level 2: Understand
CO6	Illustrate Architecture Techniques	Level 2: Understand

Mapping of course outcomes with program outcomes

COs	PROGRAMME OUTCOMES (PO)											
	K3 PO1	4 PO2	5 PO3	5 PO4	6 PO5	3 PO6	2 PO7	3 PO8	3 PO9	3 PO10	3 PO11	3 PO12
CO1	1	-	-	-	-	-	-	3	-	-	-	-
CO2	2	-	-	-	-	-	-	-	-	-	-	-
CO3	2	1	-	3	3	-	1	-	-	1	-	-
CO4	2	-	1	-	-	-	-	-	-	-	-	-
CO5	1	-	2	-	-	-	-	1	-	-	-	1
CO6	2	-	-	-	2	-	1	-	-	-	-	-

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	20	20	20
Understand	30	30	80
Apply			
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part Acontain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. List out the importance of software architecture.
2. Explain the business context of software architecture.
3. Describe the Contexts of Software Architecture.
4. Explain roles of the Software Architect.

Course Outcome 2 (CO2):

1. List out basic concepts in Software Architecture.
2. Explain the relevance of Software Design.
3. Examine the Design principles of software architecture.

Course Outcome 3 (CO3):

1. Summarize structural design patterns.

2. Illustrate Design Patterns.

Course Outcome 4 (CO4):

1. Compare and contrast waterfall model and agile model.
2. Classify architecture evaluation methods.
3. Illustrate documentation of Software Architectures with example.
4. Identify different phases in software development.

Course Outcome 5 (CO5):

1. Identify the basis for economic analysis of architecture.
2. Identify the role of Product Line Architecture.
3. Develop a strategy to improve the Architecture Competence.
4. Select the quality attributes that are relevant for the Software Industry.

Course Outcome 6 (CO6):

1. Differentiate bottom-up and top-down architecture development.
2. Write short note on architectural patterns
3. Describe Architecture Development Techniques.
4. Explain different architecture integration strategies.

Model Question paper

Course Code: **ITT385**

Course Name: Software Architecture Concepts

Max.Marks:100 Duration: 3 Hours

PART A **(10*3=30)**

(Each Question carries 3 Marks)

1. List out the importance of software architecture.
2. Explain the business context of software architecture.
3. List out basic concepts in Software Architecture.
4. Summarize structural design patterns.
5. Compare and contrast waterfall model and agile model.
6. Classify architecture evaluation methods.
7. Identify the basis for economic analysis of architecture.
8. Identify the role of Product Line Architecture.
9. Differentiate bottom-up and top-down architecture development.
10. Write short note on architectural patterns.

PART B **(5*14=70)**

(Each Question carries 14 Marks)

11. Describe the Contexts of Software Architecture.

OR

12. Explain roles of the Software Architect.
13. Analyze the relevance of Software Design.

OR

14. Examine the merits and demerits of each type ofDesign Patterns.
15. Illustrate documentation of Software Architectures with example.

OR

16. Identify different phases in software development.
17. Develop a strategy to improve the Architecture Competence.

OR

18. Select the quality attributes that are relevant for the Software Industry.
19. Describe Architecture Development Techniques.

OR

20. Explain different architecture integration strategies.

Syllabus

Module 1: Introduction to Software Architecture
Relevance of Software Architecture, Contexts of Software Architecture, Software ArchitectureApproaches, Software Architect, Roles of the Software Architect
Module 2: Basic Conceptsin Software Architecture
Basic Concepts in Software Architecture, Introduction to software design, Design principles, Design Patterns
Module 3: Architecture in SDLC
Software Development Life Cycle (SDLC) Role of Architecture in SDLC -Requirements and Design Documenting Software Architectures, Implementation, and Testing, Architecture Evaluation
Module 4: Architecture & Business
Economic Analysis of Architectures Architecture Competence- Competence of Individuals,Competence of a Software Architecture Organization Architecture and Software Product Lines- working, role, evaluation, and issues Quality Attributes
Module 5: Architecture Techniques

Architecture Development Techniques Software Partitioning Strategies Software Changeability and Dependency Management Using Architectural Patterns Integration Strategies Bottom-Up Architecture Development, Top-Down Architecture Development

Text Books

1. Software Architecture – A practical Guide using UML, Jeff Garland, Richard Anthony, John Wiley & Sons Ltd, ISBN 0 470 84849 9 , 2003
2. Software Architecture in Practice, (3rd Edition) (SEI Series in Software Engineering), by Len Bass, Paul Clements, Rick Kazman, Publisher: Addison-Wesley, 2012.
3. Software Design: From Programming to Architecture, Eric Braude, Wiley, 2004.
4. Software Architecture: Foundations,Theory, and Practice, R. N. Taylor, N. Medvidovic, and E. M. Dashofy. , John Wiley & Sons, 2009.

Reference Books

1. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3ed), Craig Larman; Printice Hall PTR (2004). ISBN 13: 978-0131489066
2. Pankaj Jalote, An integrated approach to Software Engineering, 3rd Edition, Springer/Narosa.
3. Ian Sommerville, Software Engineering, 10th Edition, Addison-Wesley

Course Contents and Lecture Schedule

Sl.No	Module 1: Introduction to Software Architecture	10hrs
1.1	Introduction to Software Architecture, importance of Software Architecture	2
1.2	Contexts of Software Architecture – Technical Context, Project Life Cycle Context, Business Context, Professional Context	2
1.3	Software Architecture Approaches - The 4+1 Views, RM-ODP viewpoints, Bass architectural structures, Hofmeister software architecture views.	2
1.4	Software Architect, Roles of the Software Architect - Relationship to other key roles in development organization, Skills and Background for the Architect,	2
1.5	Injecting Architecture Experience, Structuring the Architecture Team, Traps and Pitfalls Associated with the Role of Software Architect.	2

	Module 2: Basic Conceptsin Software Architecture	**8 hrs**
2.1	Basic Concepts in Software Architecture – Architecture, components, connector, configuration, architectural style, architectural patterns, models, processes, stakeholders.	2
2.2	Introduction to software design	1
2.3	Design principles - Correctness and Robustness, Flexibility, Reusability, and Efficiency.	2
2.4	Design Patterns - Creational Design Patterns, Structural Design Patterns, Behavioural Design Patterns.	3
	Module 3: Architecture in SDLC	**9 hrs**
3.1	**Software Development Life Cycle** (SDLC) overview – Phases in software development, Different types of SDLC – Waterfall model to Agile model.	2
3.2	**Architecture in SDLC** - Architecture and Requirements	2
3.3	**Architecture in SDLC** - Designing an Architecture	2
3.4	**Architecture in SDLC** - Documenting Software Architectures, Architecture, Implementation, and Testing, Architecture Evaluation.	3
	Module 4: Architecture & Business	**9 hrs**
4.1	**Economic Analysis of Architectures** - Decision-Making Context, The Basis for the Economic Analysis, Putting Theory into Practice: The CBAM	2
4.2	**Architecture Competence**- Competence of Individuals: Duties, Skills, and Knowledge of Architects, Competence of a Software Architecture Organization	2
4.3	**Architecture and Software Product Lines** - An Example of Product Line Variability, Working of a Software Product Line, Product Line Scope, The Quality Attribute of Variability, The Role of a Product Line Architecture, Variation Mechanisms, Evaluating a Product Line Architecture, Key Software Product Line Issues	2
4.4	**Quality Attributes** – Availability, Interoperability, Modifiability, Performance, Security, Testability, Usability	3
	Module 5: Architecture Techniques	**9 hrs**

5.1	**Architecture Development Techniques -** Commonality and variability analysis, Design for change, Generative programming techniques, Building a skeleton system, Prototyping, Interface development – Design by Contract, Architectural description languages, Architecture evaluation	2
5.2	**Software Partitioning Strategies** – Separation of Concerns - Functional decomposition, Isolate configuration data, Isolate hardware-specific components, Isolate time-critical components, Separate domain implementation model from human interface, Separate domain implementation model from implementationtechnology, Separate main function from monitoring, Separate fault recovery processing, Adaptation of external interfaces	2
5.3	**Software Changeability and Dependency Management** - The stable dependencies principle (SDP), Acyclic dependencies principle, Interface Separation Principle. **Using Architectural Patterns**	2
5.4	**Integration Strategies -** Data-only integration, Executable integration, Establishing Architecture to Support Development, Configuration and change management, Build management, Continuous integration, Anticipate multi-language development, Anticipate tactical development (scripting),	2
5.5	Bottom-Up Architecture Development, Top-Down Architecture Development	1

SEMESTER V

HONOURS

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT393	WIRELESS COMMUNICATION	VAC	3	1	0	4

Preamble: The course addresses the fundamentals of wireless communications and provides an overview of existing and emerging wireless communication technology and networks.

Prerequisite: ITT292 Mathematical Foundation for Networking

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome(CO)	Bloom's Category
CO 1	Discuss the fundamental concepts wireless communication	Level 2: Understand
CO 2	Illustrate large and small scale fading in mobile wireless communication	Level 3: Apply
CO 3	Familiarize and apply equalization, diversity & channel coding techniques	Level 3: Apply
CO 4	Identify the multiple access techniques in wireless systems	Level 2: Understand
CO 5	Discuss various wireless system models	Level 2: Understand

Mapping of course outcomes with program outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	1	2	1	-	-	-	-	-	-	-	-	3
CO 2	1	2	2	2	-	-	-	-	-	-	-	2
CO 3	1	2	3	2	-	-	-	-	-	-	-	2
CO 4	1	2	2	1	-	-	-	-	-	-	-	2
CO 5	1	2	1	-	2	-	-	-	-	-	-	3

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember			
Understand	30	30	60
Apply	20	20	40
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Define Coherence time. How does this parameter decide the behaviour of the wireless channel?
2. If a total of 33MHz of bandwidth is allocated to a particular FDD cellular telephone system which uses two 25KHz simplex channels to provide full duplex voice and control channels, compute the number of channels available per cell if a system uses a) 4 cell re-use and b) 7-cell reuse.

Course Outcome 2 (CO2)

1. Distinguish between slow and fast fading. Explain with an example.
2. Describe the free space propagation model and derive the loss in the signal strength.

Course Outcome 3(CO3):

1. Draw and explain a simplified communication system using an adaptive equalizer at the receiver.
2. Write a brief note on categories of space diversity reception methods.

Course Outcome 4 (CO4):

1. How FDMA handles near far problem?
2. Identify the channel capacity of TDMA in cell system.

Course Outcome 5 (CO5):

1. Explain the GSM architecture in detail.
2. What is triangular routing problem? Discuss any solution.

Model Question Paper

Course Code: ITT393

Course Name: WIRELESS COMMUNICATION

Max. Marks: 100 Duration: 3 Hours

PART A

Answer all questions. Each question carries 3 marks. (10 * 3 = 30 Marks)

1. How is frequency reuse distance measured in cellular system?
2. What is meant by mobile assisted handoff?
3. What is fast fading?
4. List the various path loss models for large scale fading
5. What is time diversity?
6. What is the difference between linear and non linear equilization?
7. What is reverse channel interference?
8. What is SDMA?
9. What is care of address?
10. What are the features of mobile adhoc networks?

Part B

Answer all questions. Each question carries 14 marks. (5 * 14 = 70 Marks)

11. Explain in detail about the handoff strategies used in cellular system

OR

12. Discuss the impact of interference in a cellular system and system capacity

13. Explain in detail about the three basic propagation mechanisms

OR

14. Explain the different types of small scale fading based on multipath time delay spread

15. Describe any two diversity combining techniques stating their respective merits

OR

16. What are block codes? Explain the features of block codes

17. Compare FDMA and TDMA

OR

18. Describe in detail about CSMA/CD protocol

19. Discuss the system architecture of GSM

OR

20. What is triangular routing? How can it be avoided?

Syllabus

Module 1: INTRODUCTION TO WIRELESS COMMUNICATION	**8 hours**
Introduction to wireless communication systems: Evolution of mobile radio communications, Mobile radio systems around the world. Example of wireless communication systems. Modern wireless communication systems: 2G, 3G, 4G and 5G. Wireless local loop, Wireless local area networks, Bluetooth and personal area networks. The Cellular Concept: Frequency reuse, channel assignment and handoff strategies, interference and system capacity, trunking and grade of service. Improving coverage and capacity in cellular systems.	
Module 2 :MOBILE RADIO PROPAGATION	**9 hours**
Large scale path loss: Introduction to radio wave propagation, free space propagation models, Three basic propagation mechanisms, reflection, Two-ray propagation model, Diffraction, Scattering. Small scale fading and multipath: Small scale multipath propagation, Types of small scale fading – flat fading, frequency selective fading, fast fading and slow fading.	
Module 3: EQUALIZATION, DIVERSITY & CHANNEL CODING	**9 hours**
Introduction, fundamentals of equalization, Survey of equalization techniques, Methods for Channel Diversity – Space Diversity, Polarization Diversity, Frequency Diversity , Multipath diversity, Time Diversity. Diversity Combining – Selection Combining, Scanning Combining, Equal Gain Combining, Maximal Ratio Combining Fundamentals of Channel Coding – BlockCodes, Examples	
Module 4:MULTIPLE ACCESS TECHNIQUES	**10 hours**
Introduction. Frequency Division Multiple Access (FDMA). Time Division Multiple Access	

(TDMA). Spread Spectrum Multiple Access. Space Division Multiple Access (SDMA). Capture effects in packet Radio, CSMA/CA Capacity of Cellular Systems – fundamentals	
Module 5 : WIRELESS SYSTEMS	**9 hours**
Telecommunication system – GSM, Wireless LAN – IEEE 802.11,Bluetooth. Mobile Network layer – Mobile IP, Mobile ad-hoc networks.	

Text Books

1. Theodore S. Rappaport, "Wireless Communications: Principles and Practice", 2^{nd} edition, Pearson Education India, 2014.
2. Dr Jochen Schiller, "Mobile Communications", 2^{nd} edition, Pearson Education, 2012.

Reference Books

1. Goldsmith, A. (2005). Wireless Communications. Cambridge: Cambridge University Press. doi:10.1017/CBO9780511841224.
2. Prof. Neal Patwari, ECE 5325/6325: Wireless Communication Systems Lecture Notes, Fall 2011
3. Cory Beard and William Stallings, "Wireless Communication Networks and Systems", Pearson

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**MODULE 1**	**8 hours**
1.1	Introduction to wireless communication systems: Evolution of mobile radio communications, Mobile radio systems around the world. Example of wireless communication systems.	1 Hr
1.2	Modern wireless communication systems: 2G, 3G, 4G and 5G. Wireless local loop, Wireless local area networks, Bluetooth and personal area networks.	1 Hr
1.3	The Cellular Concept: Frequency reuse, channel assignment and handoff strategies,	2 Hrs
1.4	Interference and system capacity, trunking and grade of service.	2 Hrs
1.5	Improving coverage and capacity in cellular systems.	2Hrs
2	**MODULE 2**	**9 hours**
2.1	LARGE SCALE PATH LOSS: Introduction to radio wave propagation, free space propagation models, Three basic propagation mechanisms, reflection, Two-ray propagation model.	3 Hrs

2.2	Diffraction, Scattering.	2 Hrs
2.3	SMALL SCALE FADING AND MULTIPATH: Small scale multipath propagation	1 Hr
2.3	Types of small scale fading – flat fading, frequency selective fading, fast fading and slow fading.	3 Hrs
3	**MODULE 3**	**9 hours**
3.1	Introduction, fundamentals of equalization, Survey of equalization techniques.	2 Hrs
3.2	Methods for Channel Diversity – Space Diversity, Polarization Diversity, Frequency Diversity, Multipath diversity, Time Diversity.	3 Hrs
3.3	Diversity Combining – Selection Combining, Scanning Combining, Equal Gain Combining, Maximal Ratio Combining	2 Hrs
3.4	Fundamentals of Channel Coding – BlockCodes, Examples	2 Hrs
4	**MODULE 4**	**10 hours**
4.1	Introduction. Frequency Division Multiple Access (FDMA).	1 Hrs
4.2	Time Division Multiple Access (TDMA).	1 Hr
4.3	Spread Spectrum Multiple Access.	2 Hrs
4.4	Space Division Multiple Access (SDMA).	2 Hrs
4.5	Capture effects in packet Radio, CSMA/CA	2 Hrs
4.6	Capacity of Cellular Systems - fundamentals	2 Hrs
5	**MODULE 5**	**9 hours**
5.1	Telecommunication system – GSM.	2 Hr
5.2	Wireless LAN – IEEE 802.11	2 Hr
5.3	Bluetooth.	1 Hrs
5.4	Mobile Network layer – Mobile IP.	2 Hrs
5.5	Mobile ad-hoc networks.	2 Hrs

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT395	**SECURITY IN COMPUTING**	**VAC**	**3**	**1**	**0**	**4**

Preamble:
The syllabus is designed with the view of preparing the students capable of understanding the principles and concepts of computer security. The students should be able to understand what it means for a system to be secure. Furthermore, the students will get to know about computing systems vulnerabilities, threats, and security controls.

Prerequisite: Basics of Operating systems, Database Systems and Computer Networks

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome(CO)	Bloom's Category
CO 1	Outline the basic concepts and techniques of computer security	Level 2: Understand
CO 2	Explain the various aspects of program security	Level 2: Understand
CO 3	Model secure and trusted operating systems	Level 3: Apply
CO 4	Summarize the requirements and features of database security	Level 2: Understand
CO 5	Identify the security issues in network and the appropriate security measures	Level 3: Apply

Mapping of course outcomes with program outcomes

COs	PROGRAMME OUTCOMES (PO)											
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	1			1						
CO 2	3	2	1			1						
CO 3	2	2	1		2							1
CO 4	2	2	1		2			2				2
CO 5	3	2	2		2	2		2				3

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember	10	10	20
Understand	30	30	40
Apply	10	10	40
Analyze			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Outline substitution techniques with suitable examples.
2. Distinguish between vulnerability, threat and control
3. Explain DES algorithm.

Course Outcome 2 (CO2):

1. Explain how non malicious program errors become a threat to security.
2. Explain the different methods to control program threats.
3. Explain the limitations on the amount of information leaked per second through a covert channel in multi-access computing system.

Course Outcome 3 (CO3):

1. Model the layered design of a trusted operating system.
2. Compare Simple Security Condition Preliminary version and Star Property Preliminary version in Bell-La-Padula model.
3. Write a set of rules combining the secrecy controls of the Bell-La-PAdula model with the integrity controls of Biba model.

Course Outcome 4 (CO4):

1. Outline the models for designing multilevel secure database.
2. Explain two phase update with an example.
3. Explain the factors that determine the sensitivity of data.

Course Outcome 5 (CO5):

1. Compare the different types of firewalls.
2. Explain the two general types of IDS.
3. Identify the security purpose for the fields, such as sequence number of an IPSec packet.

Model Question paper

Course Code: ITT395

Course Name: Security in Computing

Max.Marks:100 Duration: 3 Hours

PART A **(10*3=30)**

(Each Question carries 3 Marks)

1. What do you mean when you say that a system is secure?
2. Summarize the uses of encryption.
3. Differentiate the types of program security flaws.
4. Explain about virus signatures.
5. Outline how a fence register is used for relocating a user's program.
6. List few disadvantages of using physical separation in computing system.
7. Interpret the purpose of encryption in multilevel secure database management systems.
8. Explain about commutative filters.
9. Identify a counter measure for traffic flow analysis.
10. Explain the different types of Intrusion Detection Systems.

PART B **(5*14=70)**

11. a. Explain the major vulnerabilities that a computer system is subjected to (7)
    b. Differentiate substitution ciphers and transposition ciphers with examples (7)

OR

12. a. Explain DES algorithm. (7)

b. Differentiate symmetric and asymmetric encryption. (7)

13. a. Explain about non malicious program errors. (7)

b. Outline the various methods to control different program threats. (7)

OR

14. a. Explain the different kinds of malicious codes. (7)

b. Represent three controls that could be applied to detect or prevent salami attacks. (7)

15. a. According to Bell-La Padula Model, identify the restrictions placed on two active subjects that need to send and receive signals to and from each other. Justify your answer. (7)

b. Compare Simple Security Condition Preliminary version and Star Property Preliminary version in Bell-La-Padula model. (7)

OR

16. a. Model the layered design of a trusted operating system. (9)

b. Explain the factors that determine the sensitivity of data. (5)

17. a. Outline the basic security requirements of database systems. (7)

b. Show the mechanisms to implement 'seperation' in databases. (7)

OR

18. a. Represent the models for designing multilevel secure database. (7)

b. Explain the disadvantages of partitioning as means of implementing multilevel security for database. (7)

19. a. Examine the significance of dual signature in secure electronic transactions? (5)

b. Compare the different types of firewalls. (9)

OR

20. Differentiate between message confidentiality threats and message integrity threats. (7)

21. Make use of a social engineering attack to obtain a user's password and explain the attack in detail. (7)

Syllabus

Module 1: Introduction to Security in Computing (10 Hours)
Introduction: Security Problem in Computing, Elementary Cryptography- Terminology and Background, Introduction - Substitution Ciphers, Transposition Ciphers, Encryption Algorithms, DES, AES, Public Key Encryption, Uses of Encryption.
Module 2: Program Security (9 Hours)
Secure Programs, Nonmalicious Program Errors, Viruses and other Malicious Code, Targeted Malicious Code, Controls against Program Threats.
Module 3 : Protection in General Purpose Operating System (9 Hours)
Protected Objects and Methods of Protection, Memory Address Protection, Control of Access to General Objects, File Protection Mechanisms, User Authentication, Designing Trusted Operating Systems- Security Policies, Models of Security, Trusted Operating System Design, Assurance in Trusted OS.
Module 4 : Database and Data Mining Security (9 Hours)
Introduction to Databases, Security Requirements, Reliability and Integrity, Sensitive Data, Inference, Multilevel Databases, Proposals for Multilevel Security, Data Mining.
Module 5 : Security in Networks (8 Hours)
Security in Networks- Threats in Networks, Network Security Controls, Secure Electronic Transactions, Firewalls, Intrusion Detection Systems.

Text Books

1. Charles P. Pfleeger, Shari Lawrence Pfleeger and Deven N. Shah, Security in Computing, 4th Edition.
2. William Stallings, Cryptography and Network Security Principles and Practice, Pearson Education, 4th Edition.

Reference Books

1. William Stallings, Network Security Essentials, Applications and Standards, Pearson Education.
2. Michael E. Whitman and Herbert J Mattord, Principles of Information Security, 4th Edition.

Course Contents and Lecture Schedule

	Module 1: Introduction to Classical and Modern cryptographic techniques	**10 hrs**
1.1	Introduction: Security problem in Computing	1
1.2	Elementary Cryptography- terminology and background	1
1.3	Substitution Ciphers	2
1.4	Transposition Ciphers	2
1.5	Encryption Algorithms- DES, AES	3

1.6	Public Key Encryption, Uses of Encryption.	1
	Module 2: Program Security	**9 hrs**
2.1	Secure Programs	1
2.2	Nonmalicious Program Errors	1
2.3	Viruses and other Malicious Code	2
2.4	Targeted Malicious Code	3
2.5	Controls against Program Threats	2
	Module 3: Protection in General Purpose Operating System	**9 hrs**
3.1	Protected Objects and Methods of Protection	1
3.2	Memory Address Protection	1
3.3	Control of Access to General Objects	1
3.4	File Protection Mechanisms	1
3.5	User Authentication	1
3.6	Designing Trusted Operating Systems- Security Policies, Models of Security	2
3.7	Trusted Operating System Design, Assurance in Trusted OS	2
	Module 4: Database and Data Mining Security	**9 hrs**
4.1	Introduction to Databases	1
4.2	Security Requirements	1
4.3	Reliability and Integrity	1
4.4	Sensitive Data, Inference	2
4.5	Multilevel Databases, Proposals for Multilevel Security	3
4.6	Data Mining	1
	Module 5: Security in Networks	**8hrs**
5.1	Network Concepts	1
5.2	Threats in Networks	2
5.3	Network Security Controls	2
5.4	Secure Electronic Transactions	1
5.5	Firewalls	1
5.6	Intrusion Detection Systems	1

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT397	ADVANCED COMPUTER ARCHITECTURE	VAC	3	1	0	4

Preamble: Advanced computer architecture course is intended to deliver students the advanced concepts of Computer architecture It also helps them to learn how computer performance is measured and how memory organisation and memory performance optimization is done. A detailed insight into ILP,TLP and DLP, multicore and shared memory architectures with necessary case studies are also covered in the syllabus.

Prerequisite: ITT204 Computer Organisation

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome(CO)	Bloom's Category
CO 1	Measure performance of a computer by understanding the basic architectures of computers	Level 3: Apply
CO 2	Demonstrate the Memory optimization techniques.	Level 3: Apply
CO 3	Investigate pipelining techniques, ILP, multithreading and to illustrate various methods to overcome the challenges in ILP	Level 3: Apply
CO 4	Write a simple OpenMP program to execute multi threaded programs and to understand the concepts in Shared Multicore Architectures	Level 3: Apply
CO 5	Write a simple CUDA program to exploit DLP and to understand the concepts in DLP	Level 3: Apply

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1	2	2								2
CO2	3	2	2	2	3							2
CO3	3	3	2	3	3							2
CO4	3	3	3	3	2							2
CO5	3	3	3	3	2							2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Test		End Semester Examination
	1	2	
Remember	10	10	20
Understand	10	10	20
Apply	30	30	60
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):
1. Consider two programs A and B that solves a given problem. A is running on a SPARC machine operating at 500 MHz and B is running on a DEX machine running at 750 MHz. A

contains a total instructions of 4670, out of which 19% are floating point instructions, 33% load store instructions and rest are simple instructions. B is composed of 25% floating point instructions. The number of simple instructions in B is twice the count of load store instructions. Total instruction count of B is 4100. In both DEX and SPARC floating point operation has an average CPI of 5 and simple instruction has an average CPIof 1.2. Both the architectures differ in the CPI of load store instruction. They are 2 and 2.4 for SPARC and DEX respectively.
a) Given this setup which machine solves the problem faster, and by how much?
b) Is there any other program to machine mapping that gives a better result ?

2. A new floating-point system is introduced in a system and it speeds up floating point operations by 2 times. In an application 1/5 th of the instructions are floating-point operations. What is the overall speedup? (Ignore the penalty to any other instructions) The speeding up of the floating-point unit slowed down data cache accesses resulting in a 1.5x slowdown. Data cache accesses consume 10% of the execution time. What is the overall speedup now?

Course Outcome 2 (CO2):
1. Discuss various cache optimisation techniques

2. An 8KB direct-mapped write-back cache is organized as multiple blocks, each of size 32-bytes. The processor generates 32-bit addresses. The cache controller maintains the tag information for each cache block comprising of the following. 1 Valid bit 1 Modified bit As many bits as the minimum needed to identify the memory block mapped in the cache. What is the total size of memory needed at the cache controller to store meta-data (tags) for the cache?

Course Outcome 3 (CO3):
1. Assume a MIPS pipeline with 1 integer unit (EX), 1 FP Adder for the following code.

Loop: **L.D F0,)(R1)** ;F0=array element
ADD.D F4,F0,F2 ;add scalar in F2
S.D F4,0(R1) ;store result
DAADUI R1,R1,#-8 ;decrement pointer 8 bytes(per DW)
BNE R1,R2,Loop ; branch if R1!=R2

(a) What is the execution time of one loop iteration (in cycles) if operand forwarding is permitted?
(b) What is the execution time of one loop iteration (in cycles) if compiler scheduling is done within an iteration of the loop?
(c) Unroll the loop fully and then schedule the code for maximum performance. What is the best execution time you can get from this code?

2. Suppose you have the following instruction sequence to be executed
lw $1, 0($7)
addi $1, $1, 1

sw $10, 10($7)
lw $2, 0($8)
addi $2, $2, 1
sw $20, 10($8)

Rearrange the instruction sequence so that it achieves the same functionality but best performance (shortest execution time). You are only allowed to change the order of the six instructions. Do not modify or add new instructions. Calculate the execution time of the instruction sequence you rearranged.

Course Outcome 4 (CO4):

1. Discuss Cache coherence protocols
2. Create a simple OpenMP program that does the following:

   A. Creates a parallel region
   B. Has each thread in the parallel region obtain its thread id
   C. Has each thread print "Hello World" along with its unique thread id
   D. Has the master thread only, obtain and then print the total number of threads

Course Outcome 5 (CO5):

1. Assume that the processor runs at 700 MHz and has a maximum vector length of 64. The load/store unit has a start-up overhead of 15 cycles; the multiply unit, 8 cycles; and the add/subtract unit, 5 cycles. Consider the following code, which multiplies two vectors that stores single precision complex numbers in it:

```
for (i=0;i<300;i++){

        c_re[i] = a_re[i] * b_re[i] – a_im[i] * b_im[i];
        c_im[i] = a_re[i] * b_im[i] + a_im[i] * b_re[i];   }
```

(a) What is the arithmetic intensity of this kernel, if arithmetic intensity is defined as the ratio of floating point operations per byte of memory accessed?

(b) Assuming chaining and a single memory pipeline, how many chimes are required? How many clock cycles are required per complex result value, including start-up overhead? (5)

2. Write a CUDA program for matrix multiplication by exploiting DLP

Model Question Paper

Course Code: ITT397

Course Name: Advanced Computer Architecture

Max.Marks:100 Duration: 3 Hours

PART A **(10*3=30)**

(Each question carries 3 Marks)

1. Compare SISD and SIMD architectures.
2. Discuss Amdahl's law.
3. For a cache with capacity 32KB, how many blocks does the cache holds for block size=32 bytes, 64 bytes and 128 bytes?
4. Differentiate cache memory and TLB
5. List the limitations of ILP
6. What do you mean by hardware speculation?
7. What is multiprocessor hyperthreading
8. What is the use of Vector Mask Registers Vector architecture.
9. Write any three differences between Vector architecture and GPU.
10. Show that the following loop have a loop-carried dependency or not?

```
for (i=0;i<100;i++){
    A[i] = B[2*i+4];
    B[4*i+5] = A[i];}
```

PART B **(5*14=70)**

(Each full question carries 14 marks)

11. a) Explain Flynn's taxonomy of architectures specifying the application of each. (7 marks)

b) A single processor has FIT of 100. What is the mean time to failure for this system? If it takes 2 days to get the system running again, what is the availability of the system? Suppose a cluster Lucid has 1000 processors with a FIT of 100, then what is its MTTF? Assume that it experiences a catastrophic failure only if 1/4 of the computers fail. (7 marks)

OR

12. a) Consider a code fragment A=D*(B+C)-E where A, B, C, D and E are memory locations to be executed on a processor TITAN. Write down the instruction sequence generated for this code fragment if TITAN is

(i) Stack machine

(ii) Accumulator machine

(iii) Load Store machine. (6 marks)

b) A new floating-point system is introduced in a system and it speeds up floating point operations by 2 times. In an application 1/5 th of the instructions are floating-point operations. What is the overall speedup? (Ignore the penalty to any other instructions) The speeding up of the floating-point unit slowed down data cache accesses resulting in a 1.5x slowdown. Data cache accesses consume 10% of the execution time. What is the overall speedup now? (8 marks)

13. a) Explain the memory hierarchy in ARM Cortex- A8 (5 marks)

b) Discuss the cache optimisation techniques. (9 marks)

OR

14. a) Consider two cache architectures. One has a separate I and D cache of size 16KB and the other one is a unified dual ported of size 32KB. The I & D cache has instruction miss rate 0.5% and data miss rate 5%. The unified cache has aggregate miss rate 1%. Hit time is 1 cycle. Miss penalty is 50 cycles. 30% of instructions are load/store. Which one is better and what is the improvement in CPI. Assume CPI of 1 without cache misses.

(6 marks)

b) Discuss how address translation is done in Virtual memory (8 marks)

15. a) We have a program of 1000 instructions in the format of "lw, add, lw, add, ..."

The add instruction depends (and only depends) on the lw instruction right before it. The lw instruction depends (and only depends) on the add instruction right before it. If the program is executed on the pipelined datapath with 5 stages (IF-ID&DR-EXE-MEM-WB).

(1) What would be the actual CPI if operand forwarding is permitted?

(2) Without forwarding, what would be the actual CPI?

Format : LOAD Rdest, #constant(Rx)

ADD Rdest,Rsrc1,Rsrc2 (7 marks)

b) Perform Tomasulo's algorithm with reservation stations and Reorder Buffer and find out clock cycle in which the last instruction completes execution..

i) Assume the following information about functional units.

Functional unit type	**Cycles in Ex**
Integer Mul	2
Integer Div	10
Integer Add	1

ii) Assume the processor can issue into the reservation stations and reorder buffer only one instruction per cycle.

iii) Assume you have unlimited reservation stations, functional units, reorder buffer entries and CDB .

iv) The Functional units are not pipelined.

v) Fill in the cycle numbers in each pipeline stage for each instruction. For each instruction indicate where its source operands' are read from (use RF for register file, CDB for common data bus and ROB for Reorder Buffer).

vi) Also for simplicity when an operand is waiting for an execution unit's result just indicated as waiting on CDB, instead of the number of the execution unit.

vii) An instruction waiting for data on CDB can move to its EX stage in the cycle after the CDB broadcast.

viii) Assume that integer instructions also follow Tomasulo's algorithm so the result from the integer functional unit is also broadcast on CDB and forwarded to dependent instructions through CDB.

(7 marks)

OR

16. a) Discuss dynamic scheduling in Intel core i7 (9 marks)
b) A non pipelined system takes 50 ns to process a task. The same task can be processed in a six segment pipeline with a clock cycle of 10ns. Determine the speedup ratio of the pipeline for 100 tasks. What is the maximum speed up that can be achieved. ? (5 marks)
17. a) Explain shared memory multiprocessor systems (8 marks)
b) Write an OpenMP program for matrix multiplication (6 marks)

OR

18. a) Discuss Directory based cache coherency protocol (7 marks)
b) Discuss Intel Skylake processor (7 marks)
19 a) Discuss Graphical Processing Units (8 marks)
b) Write a CUDA program for adding two vectors (6 marks)

OR

20 a) Discuss Loop level parallelism with examples (8 marks)
b) Discuss DLP in Nvidia Maxwell (6 marks)

Syllabus

Module 1:(6 hours)
Introduction: Defining Computer Architecture, Flynn's Classification of Computers, Metrics for Performance Measurement-CPU performance, Memory/ Cache performance.
Module 2:(10 hours)
Memory Hierarchy Introduction, Advanced Optimizations of Cache Performance, Memory Technology and Optimizations, Virtual Memory and Virtual Machines, The Design of Memory Hierarchy, Simple program analysis using PIN (A binary instrumentation tool) Case Study: Memory Hierarchies in Intel Core i7 and ARM Cortex-A8.
Module 3:(10 hours)
Instruction Level Parallelism and Thread Level Parallelism: Introduction, Concepts and Challenges, Basic Compiler Techniques for Exposing ILP, Reducing Branch Costs with Advanced Branch Prediction, Dynamic Scheduling, Advanced Techniques for Instruction Delivery and Speculation, Limitations of ILP, Multithreading: Exploiting Thread-Level Parallelism to Improve Uniprocessor Throughput, Simple thread programs and synchronization using OpenMP, Case Study: Dynamic Scheduling in Intel Core i7 and ARM Cortex-A8
Module 4:(10 hours)
Multicore systems and Shared Memory Architectures - Introduction, Shared-Memory Multicore Systems, Performance Metrics for Shared-Memory Multicore Systems, Cache Coherence Protocols, Synchronization, Memory Consistency, Multithreaded Programming using OpenMP, Case Study:

Intel Skylake and IBM Power8.
Module 5:(9 hours)
Data Level Parallelism Introduction, Vector Architecture, SIMD Instruction Set Extensions for Multimedia, Graphics Processing Units, GPU Memory Hierarchy, Detecting and Enhancing Loop-Level Parallelism, Introduction to CUDA programming and simple programs using CUDA, Case Study: Nvidia Maxwell.

Text Books

1. J.L. Hennessy and D.A. Patterson. Computer Architecture: A Quantitative Approach. 5th Edition, Morgan Kauffmann Publishers, 2012.
2. J.P. Shen and M.H. Lipasti. Modern Processor Design: Fundamentals of Superscalar Processors. McGraw-Hill Publishers, 2005.

References

1. D.B. Kirk and W.W. Hwu. Programming Massively Parallel Processors. 2nd Edition, Morgan Kauffmann Publishers, 2012.
2. Pin tool – A Dynamic Binary Instrumentation Tool http://software.intel.com/en-us/articles/pin-a-dynamic-binary-instrumentation-tool
3. OpenMP. http://www.openmp.org/
4. CUDA. https://developer.nvidia.com/cuda-zone

Course Content and Course Schedule

Module 1:Introduction:		**6 hours**
1.1	Defining Computer Architecture	2 hours
1.2	Flynn's Classification of Computers	1 hour
1.3	Metrics for Performance Measurement	1 hour
1.4	Processor performance	1 hour
1.5	Memory Performance	1 hour
Module 2: Memory Hierarchy		**10 hours**
2.1	Introduction to Memory hierarchy	1 hour
2.2	Advanced Optimizations of Cache Performance	2 hours
2.3	Memory Technology and Optimizations	2 hours

2.4	Virtual Memory and Virtual Machines	1 hour
2.5	The Design of Memory Hierarchy	1 hour
2.6	Simple program analysis using PIN (A binary instrumentation tool)	1 hour
2.7	Case Study: Memory Hierarchies in Intel Core i7 and ARM Cortex-A8.	2 hours
	Module 3:Instruction Level Parallelism and Thread Level Parallelism	**10 hours**
3.1	Instruction-level Parallelism: Concepts and Challenges	1 hour
3.2	Basic Compiler Techniques for Exposing ILP	1 hour
3.3	Reducing Branch Costs with Advanced Branch Prediction	1 hour
3.4	Dynamic Scheduling	1 hour
3.5	Advanced Techniques for Instruction Delivery and Speculation	1 hour
3.6	Limitations of ILP	1 hour
3.7	Multithreading, Exploiting Thread-Level Parallelism to Improve Uniprocessor Throughput	1 hour
3.8	Simple thread programs and synchronization using OpenMP	1 hour
3.9	Case Study: Dynamic Scheduling in Intel Core i7 and ARM Cortex-A8	2 hours
	Module 4: Multicore systems and Shared Memory Architectures	**10 hours**
4.1	Introduction to TLP	1 hour
4.2	Shared-Memory Multicore Systems	1 hour
4.3	Performance Metrics for Shared-Memory Multicore Systems	1 hour
4.4	Cache Coherence Protocols	3 hours
4.5	Synchronization, Memory Consistency	1 hour
4.6	Multithreaded Programming using OpenMP	1 hour
4.7	Case Study: Intel Skylake and IBM Power8.	2 hours

Module 5: Data Level Parallelism		9 hours
5.1	Introduction to DLP	1 hour
5.2	Vector Architecture	1 hour
5.3	SIMD Instruction Set Extensions for Multimedia,	1 hour
5.4	Graphics Processing Units, GPU Memory Hierarchy	2 hours
5.5	Detecting and Enhancing Loop- Level Parallelism	1 hour
5.6	Introduction to CUDA programming and simple programs using CUDA	2 hours
5.7	Case Study: Nvidia Maxwell.	1 hour

SEMESTER VI

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT302	INTERNETWORKING WITH TCP/IP	PCC	3	1	0	4

Preamble: This subject is about the TCP/IP protocol suite and how it is used on the internet. It begins with a review of the underlying communications technologies needed for the internet. The course provides a detailed examination of IP routing, UDP, TCP, network virtualization, and label switching. Finally, internet applications and Software defined networking are discussed.

Prerequisite: ITT 305 Data Communication and Networking

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcomes (CO)	Bloom's Category Level
CO 1	Discuss internetworking concepts and internet address resolution.	Level2: Understand
CO 2	Illustrate the functions of IPv4, IPv6, and ICMP protocols	Level 3: Apply
CO 3	Explain internet routing architecture and internet multicasting	Level2: Understand
CO 4	Solve the design issues and protocols in transport layer	Level 3: Apply
CO 5	Explain application layer protocols, network virtualization and software defined networking	Level2: Understand

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	-	-	-	-	-	-	-	-	-	2
CO 2	3	2	2	-	-	-	-	-	-	-	-	2
CO 3	3	2	-	-	2	-	-	-	-	-	-	2
CO 4	3	2	2	-	-	-	-	-	-	-	-	2
CO 5	3	2	-	-	3	-	-	-	-	-	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test 1(Marks)	Test 2(Marks)	
Remember			
Understand	40	40	80
Apply	10	10	20

Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain ARP
2. Explain different layers in TCP/IP reference model.

Course Outcome 2 (CO2)

1. Explain IP datagram fragmentation and reassembly. Also explain different header fields affected in these cases.
2. Describe the functions of ICMP

Course Outcome 3(CO3):

1. Explain characteristics and message formats in BGP.
2. Explain RIP in detail. What is slow convergence problem and how it is solved?

Course Outcome 4 (CO4):

1. Draw and explain TCP finite state machine.

2. What is label switching?

Course Outcome 5 (CO5):

1. Explain the difference between persistent and non-persistent HTTP.
2. Explain DNS.

Model Question Paper

Course Code: ITT302
Course Name: INTERNETWORKING WITH TCP/IP

Max. Marks: 100 Duration: 3 Hours

PART A

Answer all questions. Each question carries 3 marks. (10 * 3 = 30 Marks)

1. What is internetworking?
2. Explain the role of routers in Networks.
3. Explain the header fields in IP that are used for datagram fragmentation and reassembly
4. Explain the importance of ICMP
5. What is an autonomous system?
6. Explain about IPv6 multicast address.
7. What is VPN?
8. What is the purpose of including pseudo header while computing UDP header checksum
9. What is a cookie?
10. What is persistent HTTP?

Part B

Answer all questions. Each question carries 14 marks. (5 * 14 = 70 Marks)

11 List and explain the main features of all the seven layers of the ISO/OSI reference model and compare it with TCP/IP Model.

OR

12 Explain in detail about ARP.

13 Explain the format of IPv4 and IPv6 datagram.

OR

14 Explain in detail about ICMP.

15 What is BGP? Explain the characteristics and message formats of BGP.

OR

16 What is slow convergence problem? How can it be solved?

17 Explain in detail about TCP segment format.

OR

18 What is congestion? Explain in detail about TCP congestion control.

19 Explain the working of DNS.

OR

20 Explain in detail about FTP.

Syllabus

Module 1: Introduction to Internetworking (8 Hours)
Introduction & Overview – Motivation for internetworking, TCP/IP internet, Internet Services. Internetworking Concept and Architectural Model – Introduction, Application-Level Interconnection, Network-Level Interconnection, Properties Of The Internet, Internet Architecture, Interconnection Of Multiple Networks With IP Routers. **Protocol Layering**- Introduction, The Need For Multiple Protocols, The Conceptual Layers Of Protocol Software, Functionality Of The Layers, ISO 7-Layer Reference Model, The TCP/IP 5-Layer Reference Model, Mapping Internet Addresses To Physical Addresses (ARP)
Module 2: Network Layer (8Hours)
Internet Protocol: Connectionless Datagram Delivery (IPv4, IPv6) – Introduction, Connectionless Delivery System Characteristics, Purpose And Importance Of The Internet Protocol, The IP Datagram, Datagram Type Of Service And Differentiated Services, Datagram Encapsulation, Datagram Size, Network MTU and Fragmentation, Datagram Reassembly, Header Fields Used For Datagram Reassembly, Time To Live (IPv4) And Hop

Limit (IPv6), Optional IP Items, Options Processing During Fragmentation. **Internet Protocol: Error And Control Messages (ICMP) –** Introduction, The Internet Control Message Protocol, Error Reporting Vs. Error Correction, ICMP Message Delivery, 5 Conceptual Layering, ICMP Message Format
Module 3: Routing (9 Hours)
Routing Architecture – Cores, Peers and Algorithms, Routing among Autonomous system – BGP - The Scope Of A Routing Update Protocol, Determining A Practical Limit On Group Size, Autonomous System Concept, Exterior Gateway Protocols And Reachability, BGP Characteristics, BGP Functionality And Message Types, Routing Within An Autonomous System (RIP, RIPng, OSPF, IS-IS)- Introduction, Static Vs. Dynamic Interior Routes, Routing Information Protocol (RIP), Slow Convergence Problem, Solving The Slow Convergence Problem, The Disadvantage Of Using Hop Counts, Delay Metric (HELLO), Delay Metrics, Oscillation, And Route Flapping, The Open SPF Protocol (OSPF). **Internet Multicasting** – Introduction, Hardware Broadcast, Hardware Multicast, Ethernet Multicast, The Conceptual Building Blocks Of Internet Multicast, The IP Multicast Scheme, IPv4 And IPv6 Multicast Addresses, Multicast Address Semantics, Mapping IP Multicast To Ethernet Multicast, Hosts And Multicast Delivery, Multicast Scope, Multicast Routing.
Module 4: Transport Layer (10 Hours)
Transport Layer - Transport Service, The services provided to upper layers, Transport Service primitives, UDP- Segment Structure, Remote Procedure Call, RTP, and RTCP.TCP – Service model, TCP Protocol, TCP Segment Header, Connection establishment and Release, TCP finite state machine, TCP Sliding Window, TCP timer management, Congestion Control. Label switching, flows and MPLS. Network Virtualization: VPNs, NATs, And Overlays
Module 5: Application Layer (10Hours)
Application Layer- HTTP- Overview, Persistent and non-persistent Connections, Message formats, Concept of Cookies and Web Cache -FTP - Electronic Mail– SMTP, Mail message formats, POP3, IMAP – DNS- Services provided by DNS, Overview of how DNS works, DNS Caching, Message format, DHCP.Software Defined Networking (SDN, OpenFlow)

Text Books

1.Douglas E Comer, “Internetworking with TCP/IP Principles, Protocol, and Architecture” , VolumeI, 6th Edition, Pearson Education, 2013
2.Andrew S. Tanenbaum, “Computer Networks”, Prentice Hall, 5th Edition
3.James F Kurose, Keith W Ross, Computer Networking: A top Down Approach featuring the Internet, Pearson Education, 3rd Edition

Reference Books

1. Behrouz A Forouzan, TCP/IP Protocol Suite, Fourth Edition

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction to Internetworking**	**8 Hours**
1.1	Introduction & Overview – Motivation for internetworking, TCP/IP internet, Internet Services. Internetworking Concept And Architectural Model – Introduction, Application-Level Interconnection, Network-Level Interconnection, Properties Of The Internet, Internet Architecture, Interconnection Of Multiple Networks With IP Routers.	2Hours
1.2	Protocol Layering- Introduction, The Need For Multiple Protocols, The Conceptual Layers Of Protocol Software, Functionality Of The Layers, ISO 7-Layer Reference Model.	2Hours
1.3	The TCP/IP 5-Layer Reference Model	2Hours
1.4	Mapping Internet Addresses To Physical Addresses (ARP)	2 Hours
2	**Network Layer**	**8 Hours**
2.1	Internet Protocol: Connectionless Datagram Delivery (IPv4, IPv6) – Introduction, Connectionless Delivery System Characteristics, Purpose And Importance Of The Internet Protocol, The IP Datagram, Datagram Type Of Service And Differentiated Services, Datagram Encapsulation, Datagram Size, Network MTU and Fragmentation, Datagram Reassembly, Header Fields Used For Datagram Reassembly, Time To Live (IPv4) And Hop Limit (IPv6), Optional IP Items, Options Processing During Fragmentation.	5Hours
2.2	**Internet Protocol: Error And Control Messages (ICMP)** – Introduction, The Internet Control Message Protocol, Error Reporting Vs. Error Correction, ICMP Message Delivery, 5 Conceptual Layering, ICMP Message Format	3 Hours
3	**Routing**	**9 Hours**
3.1	Routing Architecture – Cores, Peers and Algorithms, Routing among Autonomous system – BGP - The Scope Of A Routing Update Protocol, Determining A Practical Limit On Group Size, Autonomous System Concept, Exterior Gateway Protocols And Reachability, BGP Characteristics, BGP Functionality And Message Types	2 Hours
3.2	Routing Within An Autonomous System (RIP, RIPng, OSPF, IS-IS)- Introduction, Static Vs. Dynamic Interior Routes, Routing Information Protocol (RIP), Slow Convergence Problem, Solving The Slow Convergence Problem, The Disadvantage Of Using Hop Counts, Delay Metric (HELLO), Delay Metrics, Oscillation, And Route Flapping, The Open SPF Protocol (OSPF).	5Hours
3.3	Internet Multicasting – Introduction, Hardware Broadcast, Hardware Multicast, Ethernet Multicast, The Conceptual Building Blocks Of Internet Multicast, The IP Multicast Scheme, IPv4 And IPv6 Multicast Addresses, Multicast Address Semantics, Mapping	2 Hours

	IP Multicast To Ethernet Multicast, Hosts And Multicast Delivery, Multicast Scope, Multicast Routing.	
4	**Transport Layer**	**10 Hours**
4.1	Transport Layer - Transport Service, The services provided to upper layers, Transport Service primitives, UDP- Segment Structure, Remote Procedure Call, RTP, and RTCP.	2 Hours
4.2	TCP – Service model, TCP Protocol, TCP Segment Header, Connection establishment and Release, TCP finite state machine, TCP Sliding Window, TCP timer management, Congestion Control.	4 Hours
4.3	Label switching, flows and MPLS. Network Virtualization: VPNs, NATs, And Overlays	4 Hours
5	**Application Layer**	**10 Hours**
5.1	Application Layer: - HTTP- Overview, Persistent and non persistent Connections, Message formats, Concept of Cookies and Web Cache	1 Hour
5.2	FTP - Electronic Mail– SMTP, Mail message formats, POP3, IMAP	2 Hours
5.3	DNS- Services provided by DNS, Overview of how DNS works, DNS Caching, Message format	2 Hours
5.4	DHCP	1 Hour
5.5	Software Defined Networking (SDN, Openflow)	4 Hours

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT304	ALGORITHM ANALYSIS AND DESIGN	PCC	3	1	0	4

Preamble: The syllabus is prepared with a view to equip the Engineering Graduatesto learn basic concepts in algorithms, and to instil the confidence to solve non-conventional problems using different problem solving strategies.

Prerequisite:

- ITT201 Data Structures

Course Outcomes: After completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Explain asymptotic notations used in the performance analysis of algorithms and to solve recurrence equations	Level 2: Understand
CO 2	Apply divide and conquer strategy to solve practical problems efficiently	Level 3: Apply
CO 3	Apply greedy and dynamic programming techniques in algorithm design	Level 3: Apply
CO 4	Apply backtracking and branch and bound techniques in algorithm design	Level 3: Apply
CO 5	Interpret sophisticated algorithms such as string matching and approximation algorithms	Level 2: Understand

Mapping of Course Outcomes with Program Outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	-	-	-	-	-	-	-	-	-	-	2
CO 2	3	3	3	-	3	-	-	-	-	-	-	2
CO 3	3	3	3	-	3	-	-	-	-	-	-	2
CO 4	3	3	3	-	3	-	-	-	-	-	-	2
CO 5	3	3	-	-	-	-	-	-	-	-	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	Test 1 (Marks)	Test 2 (Marks)	Marks
Remember			
Understand	30	30	60
Apply	20	20	40
Analyze			
Evaluate			
Create			

Mark distribution

Total Marks	Continuous Internal Evaluation (CIE)	End Semester Examination (ESE)	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be *two* parts; **Part A** and **Part B**. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer *all* questions. Part B contains 2 questions from each module of which student should answer *any one*. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. What is an asymptotic notation? Give the different notations used to represent the complexity of algorithms.
2. Write the recurrence equation for your algorithm and solve it to estimate the time complexity of the algorithm.

Course Outcome 2 (CO 2):

1. Give the divide and conquer solution for binary search and analyse its complexity.

2. What is the notion behind the divide and conquer method? Apply divide and conquer strategy to perform merge sort on an array of integers.

Course Outcome 3 (CO 3):

1. Why Kruskal's minimum cost spanning tree construction method is considered as a Greedy method for problem solving?
2. State fractional knapsack problem. Give an algorithm for solving the fractional knapsack problem using greedy strategy.

Course Outcome 4 (CO 4):

1. Draw the state space tree corresponding to 4-Queens problem.

Course Outcome 5 (CO 5):

1. Write an algorithm based on Rabin Karp method to find all the occurrences of pattern P[0..m-1] from a given string str[0....n-1], where n>m. Compare the time complexity of this algorithm with the naive approach.
2. Suggest an algorithm for finding the vertex cover of a graph.

Model Question Paper

Course Code: ITT304

Course Name: Algorithm Analysis and Design

Max.Marks:100 **Duration: 3 Hours**

Part A

Answer all questions. Each question carries 3 marks. (10 * 3 = 30 Marks)

1. What are the properties of a good algorithm?
2. Write down the control abstraction of divide and conquer.
3. List and explain the characteristic properties associated with a problem that can be solved using dynamic programming.
4. What is backtracking? Give one problem that can be solved by backtracking.
5. Differentiate Fixed Tuple and Variable Tuple formulation.
6. Define Least Common Sequence Problem.
7. What is principle of optimality?
8. Write an algorithm for matrix multiplication using Divide and conquer method.

9. What are approximation algorithms?

10. Differentiate between deterministic and nondeterministic algorithm.

Part B

Answer all questions. Each question carries 14 marks. (5 * 14 = 70 Marks)

11. (a) Compare asymptotic notations with examples (6 marks)

(b) What is amortized analysis? Explain any one method to perform amortized analysis with an example. (8 marks)

OR

12. (a) Compare the growth of time complexity for the following set of functions. (6 marks)

(i) 2n and n^2
ii) √log n and log log n
iii) n√n and n log n

(b) Solve the recurrence equation using recursive tree method: T(n)= 2T(n/2)+c (8 marks)

13. (a) How we can prove that Strassen's matrix multiplication is advantageous over ordinary matrix multiplication? (8 marks)

(b) Explain why worst case complexity of Quick sort is $O(n^2)$ and average case complexity is O(n log n) (6 marks)

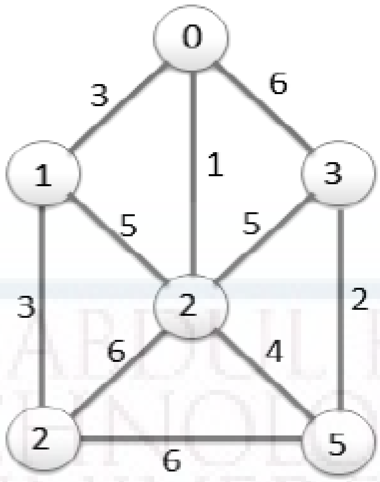
OR

14. (a) Write a recursive algorithm for implementing Binary Search. Illustrate the divide and conquer approach through this algorithm. (7 marks)

(b) Design a recursive algorithm to find the maximum and minimum from a set of *n* numbers .Illustrate with an example. (7 marks)

15. (a) Explain Kruskal's algorithm. Find the minimum cost spanning tree of the graph whose vertices are v1,v2,v3,v4,v5,v6 and v7. Cost of graph edges are (v1,v2)= 28, (v1,v6)=10, (v6,v5)=25, (v5,v4)=22, (v7,v2)=14, (v2,v3)=16, (v3,v4)=12, (v4,v7)=18 (8 marks)

(b) Draw and explain each stages of execution of Prim's algorithm in the following graph. (6 marks)



OR

16. (a) Find an optimal solution to the fractional knapsack problem for an instance with number of items 7, Capacity of the sack W=15, profit associated with the items (p1,p2,…,p7) = (10,5,15,7,6,18,3) and weight associated with each item (w1,w2,…,w7)= (2,3,5,7,1,4,1). (5 marks)
(b) Define Travelling Salesman Problem (TSP). Explain the basic steps that are to be followed to solve TSP using branch and bound. Illustrate with an example. (9 marks)

17. (a) Define the following tree organization for representing solution spaces.

(i) Problem state (ii) State space (iii) Solution state (6 marks)

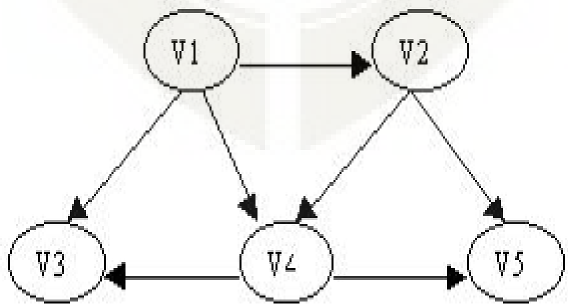
(b) Describe Branch and Bound technique. Demonstrate a problem that can be solved by branch and bound method. (8 marks)

OR

18. (a) What is 15-puzzle problem? How can it be solved? (8 marks)

(b) What is the relevance of Least cost search? Give the control abstraction for Least Cost Search. (6 marks)

19. (a) Perform Topological sorting on the given graph. (5 marks)



(b) Explain vertex cover problem using an example. Suggest an algorithm for finding vertex cover of a graph. (9 marks)

OR

20. (a) Write an algorithm based on Rabin Karp method to find all the occurrences of the pattern P[0..m-1] from a given string str[0....n-1], where n>m. Analyse and Compare the time complexity of this algorithm with the naive approach. (10 marks)

(b) Give an approximate algorithm for graph coloring problem. (4 marks)

Syllabus

Module 1: Introduction to Algorithms (9 Hours)
Properties of a good Algorithm, Development of an Algorithm, Pseudo-code Conventions, Recursive Algorithms Performance Analysis - Space and Time Complexity, Running Time Comparison - Worst, Best and Average Case Complexity, Asymptotic Notations, Common Complexity Functions Recurrence Relations – Solving Recurrences using substitution and recurrence trees Amortized Complexity – aggregate analysis, cost-accounting and potential methods
Module 2: Divide and Conquer (8 Hours)
Divide and Conquer - Control Abstraction, Finding Maximum and Minimum, Binary Search, Strassen's Matrix Multiplication, Quick Sort, Merge Sort
Module 3: Greedy Strategy and Dynamic Programming (9 Hours)
Greedy Strategy- Control Abstraction, Fractional Knapsack Problem, Minimum Cost Spanning Trees – Prim's and Kruskal's Algorithm, Job sequencing with deadlines Dynamic Programming- Principle of Optimality, DP solution for traveling salesman and 0/1 Knapsack problems, Least Common Subsequence problem
Module 4: Backtracking & Branch and Bound (10 Hours)
Backtracking– State Space Tree, Fixed Tuple and Variable Tuple Formulation - Control Abstraction, Monte Carlo Method – N-Queens Problem, Sum of Subsets Branch and Bound Techniques– FIFO, LIFO, and LC Branch and Bound, Control Abstraction, 15-puzzle problem
Module 5: Sophisticated Algorithms (9 Hours)
Topological sort, string matching: KMP algorithm, Rabin-Karp algorithm, Introduction to Computational Complexity – complexity classes, Determinism and Non-determinismm, Approximation Algorithms – Planar Graph Colouring, Vertex cover

Text Books

1. Introduction to Algorithms – Cormen, Leiserson, Rivest, Stein – 3/e, PHI
2. Fundamentals of Computer Algorithms – Horowitz and Sahni, 2/e, Universities Press

Reference Books

1. Computer Algorithms – Introduction to Design and Analysis – Sara Baase& Allen Van Gelder, 3/e, Pearson Education

2. Introduction to the Design and Analysis of Algorithms – Anany Levitin, 3/e, Pearson

3. Foundations of Algorithms – Richard Neapolitan, 5/e, Jones and Barlett Learning

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**Introduction to Algorithms**	**9 Hours**
1.1	Properties of a good Algorithm, Development of an Algorithm, Pseudo-code Conventions, Recursive Algorithms	3 Hours
1.2	Performance Analysis - Space and Time Complexity, Running Time Comparison - Worst, Best and Average Case Complexity, Asymptotic Notations, Common Complexity Functions	3 Hours
1.3	Recurrence Relations – Solving Recurrences using substitution and recurrence trees Amortized Complexity – aggregate analysis, cost-accounting and potential methods	3 Hours
2	**Divide and Conquer**	**8 Hours**
2.1	Divide and Conquer - Control Abstraction, Finding Maximum and Minimum	2 Hours
2.2	Binary Search, Strassen's Matrix Multiplication	3 Hours
2.3	Quick Sort, Merge Sort	3 Hours
3	**Greedy Strategy and Dynamic Programming**	**9 Hours**
3.1	Greedy Strategy- Control Abstraction, Fractional Knapsack Problem	2 Hours
3.2	Minimum Cost Spanning Trees – Prim's and Kruskal's Algorithm, Job sequencing with deadlines	2 Hours
3.3	Dynamic Programming- Principle of Optimality	2 Hours
3.4	DP solution for traveling salesman and 0/1 Knapsack problems, Least Common Subsequence problem	3 Hours
4	**Backtracking & Branch and Bound**	**10 Hours**
4.1	Backtracking– State Space Tree, Fixed Tuple and Variable Tuple	3 Hours

	Formulation - Control Abstraction	
4.2	Monte Carlo Method – N-Queens Problem, Sum of Subsets	2 Hours
4.3	Branch and Bound Techniques– FIFO and LIFO	2 Hours
4.4	LC Branch and Bound, Control Abstraction, 15-puzzle problem	3 Hours
5	**Sophisticated Algorithms**	**9 Hours**
5.1	Topological sort, string matching: KMP algorithm	3 Hours
5.2	Rabin-Karp algorithm, Introduction to Computational Complexity – complexity classes, Determinism and Non-determinismm	3 Hours
5.3	Approximation Algorithms – Planar Graph Colouring, Vertex cover	3 Hours

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT306	DATA SCIENCE	PCC	3	1	0	4

Preamble: This course is designed to provide learners with working knowledge of the theoretical background of various aspects of Data Science and enable them to incorporate and apply the principles of statistics and machine learning to solve real-world problems for large-scale data analysis.

Prerequisites:

- MAT 101 Linear Algebra and Calculus
- MAT 208 Probability and Statistics and Advanced Graph Theory
- ITT 205 Problem Solving Using Python
- ITT 201 Data Structures
- ITT 206 Database Management Systems

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome(CO)	Bloom's Category Level
CO 1	Explain the fundamental concepts and various aspects of data science	Level 2: Understand
CO 2	Choose data validationtechniques suitable for statistical analysis andpresent results using data visualization techniques.	Level 2: Understand
CO 3	Identify different statistical learning algorithm for solving a problem	Level 3: Apply
CO 4	Use statistical analysis to characterize and interpret data sets	Level 3: Apply
CO 5	Compare the pros/cons of various models and algorithms used for data analysis and data mining	Level 2: Understand
CO 6	Develop the ability to perform basic data analysis in Python and understand the fundamentals of deep learning.	Level 3: Apply

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	1	1	1	1	2	-	-	-	-	-	-	-
CO 2	3	2	1	1	3	-	-	-	-	2	-	-
CO 3	3	2	1	1	3	-	-	-	1	2	-	-
CO 4	3	3	2	1	3	-	-	-	1	2	-	-

CO 5	2	3	1	1	3	-	-	-	1	2	-	-
CO 6	3	2	1	1	3	-	-	-	1	2	-	-

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test1 (Marks)	Test2 (Marks)	
Remember	10	10	20
Understand	25	25	50
Apply	15	15	30
Analyse			
Evaluate			
Create			

Marks distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer anyone. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO 1):

1. What is data science? What are the different models for data science?
2. Explain data science process with a neat diagram.
3. Explain different types of Data Sets in Data Science

Course Outcome 2 (CO 2):

1. List any four two tools for data visualisation?

2. What is data visualization and what are the different techniques used for visualizing data?
3. Discuss methods of evaluating models in data science?

Course Outcome 3(CO 3):

1. Explain random forest ensemble method with an example.
2. What is data cleaning? What are the different operations in data cleaning?
3. Is regression a supervised learning technique? Justify your answer. Compare it with classification giving examples.
4. What are ensemble methods? Explain the bagging technique
5. Discuss Linear discriminant analysis.
6. What is decision tree? Explain the working of decision tree with information gain algorithm.

Course Outcome 4 (CO 4):

1. Differentiate between supervised and unsupervised learning techniques.
2. Classify different types of clustering. What are the practical issues in clustering?
3. Summarise different kernel tricks in SVM.
4. Illustrate with examples different Resampling methods.
5. Suppose that our task is to cluster data points into two clusters. Let the data points are {2, 4, 10, 12, 3, 20, 30, 11, 25}. Let 2 and 4 are initial cluster centroids. Apply Two rounds of k-means algorithm and find a set of clusters. Use Euclidean distance as the measure.

Course Outcome 5 (CO 5):

1. Compare Apriori and FP Growth algorithm. What are the advantages of FP Growth over Apriori algorithm?
2. How will you relate constraint-based mining with frequent pattern mining?

3. A database has five transactions. Let min_sup=60% and min_conf=80%. With the following transaction, list all the strong association rules.

   T100 {M, O, N K, E, Y}

   T200 {D, O, N, K, E, Y}

   T300 {M, A, K, E}

   T400 {M, U, C, K, Y}

   T500 {C, O, O, K, I, E}

Course Outcome 6 (CO 6):

1. Write an example of multiplying three dimensional matrices in NumPy.
2. Identify the essential libraries in Python.
3. Is Jupyter notebook IDE?How can you relate IPython and Jupyter?
4. What are the ways to store text data in pandas?

Model Question Paper

Course Code: ITT306

Course Name: Data Science

Max.Marks:100 **Duration: 3 Hours**

Part A

Answer all questions. Each question carries 3 marks. (10 * 3 = 30 Marks)

1. What is data science? What are the different models for data science?
2. What is data visualization and what are the different techniques used for visualizing data?
3. Is regression a supervised learning technique? Justify your answer. Compare it with classification giving examples.
4. Explain random forest ensemble method with an example.
5. Explain different types of clustering. What are the practical issues in clustering?
6. What is Support Vector Machine? How classification is done using SVM?
7. Explain the concept of constraint-based mining.
8. Compare Apriori and FP Growth algorithm. What are the advantages of FP Growth over Apriori algorithm?
9. Briefly explain the essential libraries in Python.
10. What makes deep learning deep? What are the different deep learning techniques?

Part B

Each question set carries 14 marks (5 * 14 = 70 Marks)

11. Explain data science process with a neat diagram.
12. Describe data science classification with a neat diagram.

OR

13. What is data cleaning? What are the different operations in data cleaning?
14. Explain different types of Data Sets in Data Science.

15. What is decision tree? Explain the working of decision tree with information gain algorithm.
16. What are ensemble methods? Explain the bagging technique.

OR

17. Differentiate supervised and unsupervised learning techniques with examples.
18. Discuss Linear discriminant analysis.

19. Explain different types of Resampling methods.
20. What is SVM? Explain Different kernel tricks in SVM.

OR

21. Write a short note on Maximal Margin Hyperplanes. (MMH).
22. Suppose that our task is to cluster data points into two clusters. Let the data points are {2, 4, 10, 12, 3, 20, 30, 11, 25}. Let 2 and 4 are initial cluster centroids. Apply Two rounds of k-means algorithm and find a set of clusters. Use Euclidean distance as the measure.

23. Consider the transaction database given below. Set minimum support count as 2 and minimum confidence threshold as 70%.
    a) Find the frequent item-set using FP Growth Algorithm.
    b) Generate strong association rules.

Transaction ID	List of Item_Ids
T100	I1,I2,I5
T200	I2,I4
T300	I2,I3
T400	I1,I2,I4
T500	I1,I3
T600	I2,I3
T700	I1,I3
T800	I1,I2,I3,I5
T900	I1,I2,I3

24. Explain Multi-level and multi-dimensional pattern mining.

OR

25. What is data mining? Explain the process of Knowledge discovery from database.
26. A database has five transactions. Let min_sup=60% and min_conf=80%. With the following transaction, list all the strong association rules.

T100 {M, O, N K, E, Y}

T200 {D, O, N, K, E, Y}

T300 {M, A, K, E}

T400 {M, U, C, K, Y}

T500 {C, O, O, K, I, E}

27. What are the basic universal functions in Numpy?
28. What are the applications of deep learning?

OR

29. Write an example of multiplying three dimensional matrices in NumPy.

30. What are the ways to store text data in pandas?

Syllabus

Module 1: Foundations Data Science, process, and tools (9 Hours)
Introduction to data science, properties of data, asking interesting questions, classification of data science, data science process, collecting, cleaning and visualizing data, languages, and models for data science
Module 2: Statistical machine learning: introduction, regression, and classification, decision tress, random forests (11 Hours)
Introduction to statistical machine learning, parametric and non-parametric methods, supervised vs. unsupervised learning, regression and classification, linear discriminant analysis, decision trees, random forests, and bagging
Module 3: Unsupervised learning, support vector machines and resampling (9 Hours)
Principal Component Analysis, clustering algorithms, practical issues in clustering, support vector classifiers and support vector machines, resampling methods: cross-validation and bootstrapping
Module 4: Data mining, pattern mining and association rule mining (9 Hours)
Data and pattern mining, types, issues, mining frequent patterns and associations, apriori and FP growth algorithms, multi-level association mining, constraint-based mining, pruning pattern space and data space
Module 5: Python for Data Analysis, Deep learning (7Hours)
Using Python for data analysis, essential python libraries, IPython, Jupyter notebook, NumPy basics, working with pandas, deep learning methods.

Textbooks

1. Kotu, V., & Deshpande, B. (2019). Data science: Concepts and practice., Morgan Kaufmann.
2. Skiena, S. S. (2017). The data science design manual., Springer.
3. James, G., Witten, D., Hastie, T., Tibshirani, R. (2017). An Introduction to Statistical Learning: with Applications in R., Springer.
4. Han, J., Kamber, M. & Pei, J. (2012). Data mining concepts and techniques, Morgan Kaufmann.

5. McKinney, W. (2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython. Beijing: O'Reilly.

Reference Books

1. Montgomery, D. C., Runger, G. C. (2017). Applied Statistics and Probability for Engineers. John Wiley and Sons.
2. Provost, F., Fawcett, T. (2013). Data Science for Business. Beijing: O'Reilly
3. Igual, L., Seguí, S. (2017). Introduction to Data Science - A Python Approach to Concepts, Techniques and Applications. Springer.

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Foundations Data Science, process, and tools**	**9 Hours**
1.1	What is Data Science, relation with AI and machine learning (1.1, 1.2 of Kotu, V., & Deshpande, B. (2019). *Data science: Concepts and practice.*, Morgan Kaufmann.)	1 Hour
1.2	Case for Data Science, Data science classification (1.3, 1.4 of Kotu, V., & Deshpande, B. (2019). *Data science: Concepts and practice.*, Morgan Kaufmann.)	1 Hour
1.3	Properties of data, asking interesting questions (1.1, 1.2 and 1.3 of Skiena, S. S. (2017). *The data science design manual.*, Springer.)	1 Hour
1.4	Data Science process: preparation, modelling, and application (2.1, 2.2, 2.3 and 2.4 of Kotu, V., & Deshpande, B. (2019). *Data science: Concepts and practice.*, Morgan Kaufmann.)	2Hours
1.5	Collecting and cleaning data (3.2 and 3.3 of Skiena, S. S. (2017). *The data science design manual.*, Springer.)	1 Hour
1.6	Visualizing data (6.1, 6.2 and 6.3 of Skiena, S. S. (2017). *The data science design manual.*, Springer.)	1Hour
1.7	Languages and models for Data Science, evaluating models (3.1, 7.2, 7.3 and 7.4 of Skiena, S. S. (2017). *The data science design manual.*, Springer.)	2 Hours
2	**Statistical machine learning: introduction, regression, and classification, decision tress, random forests** **(Reference Textbook for all topics:** James, G., Witten, D., Hastie, T., Tibshirani, R. (2017). *An Introduction to Statistical Learning: with Applications in R.*, Springer.)	**11 Hours**
2.1	What is statistical learning, parametric and non-parametric methods (2.1)	1 Hour
2.2	Supervised vs. unsupervised learning, Classification vs. regression (2.1)	1 Hour
2.3	Simple linear regression, assessing model accuracy (3.1)	1Hour
2.4	Multiple linear regression, some important concerns (3.2)	1 Hour

2.5	Extensions of the linear model (3.3.2)	1 Hour
2.6	Classification (4.1)	1 Hour
2.7	Logistic regression: model, estimating coefficients, predicting (4.3.1, 4.3.2, 4.3.3)	2 Hours
2.8	Linear discriminant analysis, using Bayes' theorem for classification, case when p=1 (4.4.1, 4.4.2, 4.4.3)	1 Hour
2.9	Decision tress, regression and classification trees, tress vs. linear models, advantages, and disadvantages (8.1)	1 Hour
2.10	Bagging, random forests (8.2.1, 8.2.2)	1 Hour
3	**Unsupervised learning, support vector machines and resampling** **(Reference Textbook for all topics:** James, G., Witten, D., Hastie, T., Tibshirani, R. (2017). *An Introduction to Statistical Learning: with Applications in R*., Springer.)	**9Hours**
3.1	Challenge of unsupervised learning, principal component analysis (10.1, 10.2.1)	1 Hour
3.2	Clustering techniques: k-means, hierarchical (10.3.1, 10.3.2)	1Hour
3.3	Practical issues in clustering (10.3.3)	1 Hour
3.4	Overview of the support vector classifier, hyperplane, maximal margin classifier (9.1.1, 9.1.2, 9.1.3)	2 Hours
3.5	Support vector classifiers: overview and details (9.2.1, 9.2.2)	1 Hour
3.6	Support vector machines: Classification with non-linear decision boundaries (9.3.1, 9.3.2)	1 Hour
3.7	Resampling: cross-validation and bootstrapping (5.1 and 5.2)	2 Hours
4	**Data mining, pattern mining and association rule mining(Reference Textbook for all topics:** Han, J., Kamber, M. & Pei, J. (2012). Data mining concepts and techniques, Morgan Kaufmann.)	**9 Hours**
4.1	Data mining, kinds of data that can be mined (1.2, 1.3,)	1 Hour
4.2	Pattern mining: class description, mining frequent patterns and associations, classification, and regression for predictive analysis (1.4.1, 1.4.2, 1.4.3)	1 Hour
4.3	Cluster analysis, outlier analysis (1.4.4, 1.4.5), measures of pattern interestingness (1.4.6), Issues in data mining (1.7)	1 Hour
4.4	Mining frequent patterns: market basket analysis, frequent and closed item sets, association rules (6.1.1, 6.1.2)	1 Hour
4.4	Apriori algorithm, generating rules, improving efficiency (6.2.1, 6.2.2, 6.2.3)	1 Hour
4.5	FP growth algorithm (6.2.4)	1 Hour
4.6	Multi-level and multi-dimensional pattern mining (7.2.1, 7.2.2)	1 Hour
4.7	Mining quantitative association rules (7.2.3) mining rare and negative patterns (7.2.4)	1 Hour
4.8	Constraint-based mining: meta-rule guided mining (7.3.1) pattern generation, pruning pattern space and data space (7.3.2)	1Hour

5	**Python for Data Analysis, Deep learning**	**7Hours**
5.1	Why Python for data analysis? Essential libraries (1.2 and 1.3 of McKinney, W. (2017). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. Beijing: O'Reilly.)	1 Hour
5.2	IPython basics and Jupyter notebook (2.2 of McKinney, W. (2017). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. Beijing: O'Reilly.), demo of appropriate examples	1 Hour
5.3	NumPy basics, universal functions, array-oriented programming, mathematical and statistical methods, file I/O, linear algebra (4.1. 4.2, 4,3, 4.4, 4.5 of McKinney, W. (2017). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. Beijing: O'Reilly.)	2 Hours
5.4	Pandas basics, essential functionality, summarizing and computing descriptive statistics (5.1, 5.2, 5.3 of McKinney, W. (2017). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. Beijing: O'Reilly.)	2 Hours
5.5	Deep learning: networks and depth, back propagation, word and graph embeddings (11.6.1, 11.6.2, 11.6.3 of Skiena, S. S. (2017). *The data science design manual*., Springer.)	1Hour

Preamble: The course Comprehensive Course work is designed to assess the knowledge gained by the students in the core courses in the B Tech programme in Information Technology. The core subjects identified in the area of study is listed in the Prerequisite section of the syllabus. The course shall have an objective type written test of 50 marks similar to comprehensive examination like GATE. The pass minimum for this course is 25. The course will help the students in preparing for comprehensive examinations and improve the confidence in answering questions in objective mode. The course will be mapped to a faculty. The hour allotted for the course may be used by the students for practicing questions in core courses, library reading and for oral assessment if needed.

Prerequisite:

The students must have gone through the following courses before attending the comprehensive examination.

1. ITT 201 Data Structures
2. ITT 202 Principles of Object Oriented Techniques
3. ITT 206 Database Management Systems
4. ITT 303 Operating System Concepts
5. ITT 305 Data Communication and Networking

Course Outcomes: After the completion of the course the student will be able to

CO No.	**Course Outcome (CO)**	**Bloom's Category Level**
CO 1	Explain the core concepts in the courses listed in the prerequisite section (ITT 201, ITT 202, ITT 206, ITT 303, ITT 305).	Level

Mapping of course outcomes with program outcomes

POs COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	-	-	2	-	-	1	-	-	1	-	3
CO 2	3	-	-	2	-	-	1	-	-	1	-	3
CO 3	3	1	1	-	-	2	-	-	-	-	-	-

3/2/1: high/medium/low

Assessment Pattern

Mark distribution

Total Marks	CIE	ESE	ESE Duration
50	-	50	1 Hour

End Semester Examination Pattern:

Written examination will be conducted by theUniversity atthe end of the sixth semester. The written examination will be of objective type similar to the competitive examination like GATE.Syllabus for the comprehensive examination consists of 5 modules based on following five core courses in the curriculum.

1. ITT 201 Data Structures
2. ITT 202 Principles of Object Oriented Techniques
3. ITT 206 Database Management Systems
4. ITT 303 Operating System Concepts
5. ITT 305 Data Communication and Networking

The written test will be of 50 marks with 50 multiple choice questions (10 questions from each module) with 4 choices of 1 mark each covering all the five core courses. There will be no negative marking. The pass minimum for this course is 25. The course should be mapped with a faculty and classes shall be arranged for practicing questions based on the core courses listed above.

Written examination : 50marks

Total : 50 marks

Course Level Assessment and Sample Questions:

1. Consider the following sequence of operations and Find the value of S + R =?

Stack	Queue
push (52);	enqueue (43);
push (56);	enqueue (24);
pop ();	dequeue ();
push (62);	enqueue (57);
S = pop ()	R = Dequeue ()

A. 75

B. 86

C. 119

D. None of the above

2. Consider the following statements.

S1: The sequence of procedure calls corresponds to a preorder traversal of the activation tree.

S2: The sequence of procedure returns corresponds to a postorder traversal of the activation tree.

Which one of the following options is correct?

A. S1 is true and S2 is false

B. S1 is false and S2 is true

C. S1 is true and S2 is true

D. S1 is false and S2 is false

3. A full binary tree with n non-leaf nodes contains

A. $\log_2 n$ nodes

B. n+1 nodes

C. 2n nodes

D. 2n+1 nodes

4. Which one of the following are essential features of an object-oriented programming language?

(i) Abstraction and encapsulation

(ii) Strictly-typedness

(iii) Type-safe property coupled with sub-type rule

(iv) Polymorphism in the presence of inheritance

A. (i) and (ii) only

B. (i) and (iv) only

C. (i), (ii) and (iv) only

D. (i), (iii) and (iv) only

5. Which of the following is associated with objects?

A. State

B. Behavior

C. Identity

D. All of the above

6. Consider the following two statements:

S1: A publicly derived class is a subtype of its base class.

S2: Inheritance provides for code reuse.

A. Both the statements S1 and S2 are correct.

B. Neither of the statements S1 and S2 are correct

C. Statement S1 is correct and S2 is incorrect

D. Statement S1 is incorrect and S2 is correct.

7. Consider the following statements S1 and S2 about the relational data model

S1 : A Relation schema can have at most one foreign key

S2 : A foreign key in a relational schema R cannot be used to refer to tuples of R

Which of the following choices is correct?

A. Both S1 and S2 are true

B. S1 is true and S2 is false

C. S1 is false and S2 is true

D. Both S1 and S2 are false

8. Which among the following is false?

A. A relation scheme can be in Third Normal Form but not in BCNF

B. Every BCNF relation scheme is in Third Normal Form.

C. BCNF provides freedom from insertion and deletion anomalies.

D. If a relation scheme has partial dependencies, it is in Second Normal Form.

9. Consider following schedules involving three transactions:

S1 : R2(A); W2(A); R3(C); W2(B); W3(A); W3(C); R1(A); R1(B); W1(A); W1(B)

S2 : R2(A); R3(C); W3(A); W2(A); W2(B); W3(C); R1(A); R1(B); W1(A); W1(B)

S3: R1(A); R2(A); R3(B); W1(A); R2(C); R2(B); W2(B); W1(C)

Which of the above schedules is/are NOT conflict serializable?

A. Only S2

B. S1 and S3

C. Only S3

D. S2 and S3

10. Which of the following statement(s) is/are correct in the context of CPU scheduling?

A. Turnaround time includes waiting time

B. The goal is to only maximize CPU utilization and minimum throughput

C. Round-robin policy can be used even when the CPU time required by each of the processes is not known apriori

D. Implementing preemptive scheduling needs hardware support

11. Consider the following statements about process state transitions for a system using preemptive scheduling.

I. A running process can move to ready state.

II. A ready process can move to running state.

III. A blocked process can move to running state.

IV. A blocked process can move to ready state.

Which of the above statements are TRUE ?

A. I, II, and III only

B. II and III only

C. I, II, and IV only

D. I, II, III and IV only

12. Consider the statements S1 to S4

S1: Paging incurred memory overhead

S2 : Multilevel paging is necessary for pages with different size

S3 : Page size has no impact on internal fragmentation.

S4 : Paging help solve the issue of external fragmentation

Which of the above statement(s) are true? Select the appropriate options among the following.

A. S1 and S2

B. S1 and S4

C. S3 and S4

D. S3 only

13. Ten signals, each requiring 3 KHz, are multiplexed on to a single channel using FDM. How much minimum bandwidth is required for the multiplexed channel? Assume that the guard bands are 200 Hz wide.

A. 30,000 Hz

B. 31,800 Hz

C. 32,000 Hz

D. None of the above

14. In an IPv4 packet, the value of HLEN is 15, and the value of the total length field is 0X0064. How many bytes of data are being carried by this packet?

    A. 85 bytes

    B. 49 bytes

    C. 40 bytes

    D. 20 bytes

15. Consider the three-way handshake mechanism followed during TCP connection establishment between hosts P and Q. Let X and Y be two random 32 bit initial sequence numbers chosen by P and Q respectively. Suppose P sends a TCP connection request message to Q with a TCP segment having SYN bit = 1, SEQ number = X and ACK bit = 0. Suppose Q accepts the connection request. Which of the following choices represents the information Present in the TCP segment header that is sent by Q to P?

    A. SYN bit =1, SEQ number = X+1, ACK bit = 0, ACK number = Y, FIN bit = 0

    B. SYN bit =0, SEQ number = X+1, ACK bit = 0, ACK number = Y, FIN bit = 1

    C. SYN bit =1, SEQ number = Y, ACK bit = 1, ACK number = X+1, FIN bit = 0

    D. SYN bit =1, SEQ number = Y, ACK bit = 1, ACK number = X, FIN bit = 0

Syllabus

Module 1: Data Structures
Data Structures-Introduction and Overview- **Arrays**- Searching and Sorting, **Linked lists** - singly linked list, Doubly linked list, Circular linked list, **Stack**, Applications of stacks, **Queue**, **Trees -** Binary Tree, Binary Tree Traversals, **Graph**, **Hash Tables**- closed hashing and Open Hashing
Module 2: Principles of Object Oriented Techniques
The Object-Oriented Approach – Characteristics of Object-Oriented Languages - Objects & Classes – Inheritance – Reusability - Creating New Data Types - Polymorphism and Overloading - Classes fundamentals, objects, methods, constructors, Overloading Methods, Overloading Constructors, Derived Class and Base Class, Usage of super keyword, Creating a Multilevel Hierarchy, Method Overriding, Using Abstract Classes, **Exception handling-** Exception Types, Using try and catch in Java, Java's Built-in Exceptions, Creating

Exception subclasses, the Java Thread Model, Event Handling-delegation event model,.
Module 3: Database Management Systems
Fundamentals of Database Management Systems (DBMS), Database System Architecture, **Entity-Relationship Model-**ER Diagrams, **Relational Model**- Relational Schema, Relational Algebra Operations, **SQL**, **Normalization**, **Transaction Processing-**Concurrency Control
Module 4: Operating System Concepts
Operating Systems- types, System kernel, Shell, **Processes**- . Process Scheduling methods, Inter process Communication,**CPU scheduling** -scheduling algorithms,**Dead locks** -conditions for deadlock, prevention, avoidance, detection, recovery from dead lock, resource trajectories –starvation,**Memory management -**fixed &variable partitions, paging & segmentation - virtual memory concepts - demand paging - page replacement - Disk scheduling, **File system concepts**
Module 5: Data Communication and Networking
Channel capacity, **Transmission media**- Synchronous and Asynchronous transmission, **Sampling theorem**, **Data Encoding** -Encoding digital data into digital signal, Encoding analog data into digital signals, **Multiplexing**, **Error Detecting and correcting codes** , **Concept of layering -** OSI and TCP/IP Protocol Stacks; Basics of packet, circuit and virtual circuit-switching; **Data link layer-** framing, error detection, Medium Access Control, **Routing protocols -** shortest path, flooding, distance vector and link state routing; **IP addressing & fragmentation** - IPv4, CIDR notation, **IP support protocols** -ARP, DHCP, ICMP, Network Address Translation (NAT); **Transport layer**- flow control and congestion control, UDP, TCP, **Application layer protocols -** DNS, HTTP, Email.

Reference Books

1. Samanta D., Classic Data Structures, Prentice Hall India, 2/e, 2009.
2. Ellis horowitz, Sartaj Sahni, Fundamentals of Data structures, GalgotiaBooksource
3. Herbert Schildt, Java: The Complete Reference, 8/e, Tata McGraw Hill, 2011.
4. Bahrami A., Object Oriented Systems Development using the Unified Modeling Language, McGraw Hill, 1999.
5. Elmasri R. and S. Navathe, Database Systems: Models, Languages, Design and Application Programming, Pearson Education, 2013.
6. Silberschatz A., H. F. Korth and S. Sudarshan, Database System Concepts, 6/e, McGraw Hill, 2011.
7. William Stallings, Operating Systems,6th Edition,Pearson,2009
8. Andrew S. Tanenbaum, “Modern Operating Systems”, 3/e, Prentice Hall

9. J. L. Peterson and A. Silberschatz , Operating System Concepts, 8/e, Addison Wesley.
10. Behrouz A. Forouzan, Data Communications and Networking, 4/e, Tata McGraw Hill.
11. William Stallings, Computer Networking with Internet Protocols, Prentice-Hall, 2004.
12. Behrouz A. Forouzan, TCP/IP Protocol Suite, 4/e, Mc Graw Hill

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITL332	COMPUTER NETWORKS LAB	PCC	0	0	3	2

Preamble: The course aims to equip students to perform networking using IPv4 and Ipv6. The lab covers static, default, and dynamic routing, setting up layer 2 switching, VLANs and security and access list.

Prerequisite:ITT305 Data Communication and Networking

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome(CO)	Bloom's Category Level
CO 1	Demonstrate internetworking and network components	Level 2: Understand
CO 2	Explain IPv4 addressing, IPv6 addressing, subnetting and design networks	Level 2: Understand
CO 3	Experiment with static, dynamic and inter VLAN routing	Level 3: Apply
CO 4	Make use of standard and extended access lists	Level 3: Apply
CO 5	Use Webserver, remote login, file transfer and automatic network configuration protocols	Level 3: Apply
CO 6	Use network simulators	Level 3: Apply

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	1	-	-	2	-	-	-	-	-	-	1
CO 2	3	3	3	-	-	-	-	-	-	-	-	2
CO 3	3	3	-	-	2	-	-	-	-	-	-	2
CO 4	3	3	3	-	2	-	-	-	-	-	-	2
CO 5	3	2	-	-	-	-	-	-	-	-	-	2
CO 6	3	1	2	-	3	-	-	-	-	-	-	2

3/2/1: high/medium/low

Assessment Pattern

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	75	75	2.5 hours

Continuous Internal Evaluation Pattern:

Attendance : 15 marks
Continuous Assessment : 30 marks
Internal Test (Immediately before the second series test) : 30 marks

End Semester Examination Pattern: The following guidelines should be followed regarding award of marks

(a) Preliminary work : 15Marks
(b) Implementing the work/Conducting the experiment : 10 Marks
(c) Performance, result and inference (usage of equipments and troubleshooting): 25 Marks
(d) Viva voce : 20 marks
(e) Record : 5 Marks

General instructions: Practical examination to be conducted immediately after the second series test covering entire syllabus given below. Evaluation is a serious process that is to be conducted under the equal responsibility of both the internal and external examiners. The number of candidates evaluated per day should not exceed 20. Students shall be allowed for the University examination only on submitting the duly certified record. The external examiner shall endorse the record.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Differentiate between a router and a switch
2. Crimp a network cable

Course Outcome 2 (CO2)

1. An IP address of 172.16.0.0/16 is assigned to an ISP. The ISP has to distribute it among 17 organizations. Design the subnets
2. You are given the IP Address of 193.103.20.0 /24 and need 50 Subnets. How many hosts per network, and total networks do you get once sub netted?

Course Outcome 3(CO3):

1. An organization with 7 departments is assigned an IP address of 200.0.0.0/24. The organization should assign address to each department. Design the subnets and connect the first and third network using RIP
2. Subnet the Class B IP Address 130.13.0.0 into 500 Subnets. What is the new Subnet Mask and what is the Increment? Connect the 6^{th} and 15^{th} Subnet using static routing

Course Outcome 4 (CO4):

1. Connect hosts on the networks 17.10.0.0/8 and 168.18.0.0/16 and block FTP traffic from the first network to the second network

Course Outcome 5 (CO5):

1. Install and configure any popular webserver
2. Configure TELNET, login to a remote machine and view the files on the remote machine
3. Configure FTP and transfer files between two machines

Course Outcome 6 (CO6):

1. Implement a mesh network in NS3 network simulator. Perform RIP routing between the nodes in the network. Analyse the packet transfer between the nodes.

LIST OF EXPERIMENTS

(All the listed experiments are mandatory)

I. Internetworking Basics

1. Familiarization of Internetworking - Network Cables- Colour coding - Crimping. Internetworking Operating Systems- Configurations.
2. Backing up and restoring IoS
3. Familiarization of network components – Hub, Switch, Bridge, Router, Access Point

II. Addressing

1. Configure and verify IPv4 addressing and sub netting
2. Configure and verify IPv6 addressing and prefix
3. Compare IPv6 address types

III. IP Routing

1. Configure and verify IPv4 routing
   a. Static Routing
   b. Dynamic Routing – RIP, OSPF, EIGRP
2. Implement Unicast IPv6 Addresses on Routers and verify
   a. Static routing
   b. Dynamic routing – RIPng, OSPFv3
3. Configure and verify dual stack routing

IV. Switching Concepts

1. Layer 2 Switching Configuration – VLAN Configuration
2. VTP Configuration, VTP Pruning
3. Implement inter-VLAN routing

V. Configuring Protocols

1. HTTP
2. TELNET

3. FTP
4. DHCP

VI. Security

1. Standard Access List
2. Extended Access List
3. Familiarization of Wireshark

VII. Network Simulators

1. Familiarize with any popular network simulator

Reference Books

1. CCNA 200-301 Official Cert Guide, Volume 1,Wendell Odom, Cisco Press
2. CCNA –Cisco Certified Network Associate. Study Guide ,Todd Lammle, CCSI, Wiley India Edition-Sixth Edition

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITD 334	MINIPROJECT	PWS	0	0	3	2

Preamble: This course is designed for enabling the students to apply the knowledge to address the real-world situations/problems and find solutions. The course is also intended to estimate the ability of the students in transforming theoretical knowledge studied as part of the curriculum so far in to a working model of a software system. The students are expected to design and develop a software/hardware project to innovatively solve a real-world problem.

Prerequisites: Subjects studied up to sixth semester.

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Make use of acquired knowledge within the selected area of technology for project development.	Level 3: Apply
CO 2	Identify, discuss and justify the technical aspects and design aspects of the project with a systematic approach.	Level 3: Apply
CO 3	Interpret, improve and refine technical aspects for engineering projects.	Level 3: Apply
CO 4	Associate with a team as an effective team player for the development of technical projects.	Level 3: Apply
CO 5	Report effectively the project related activities and findings.	Level 2: Understand

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3	3	3	3	3	3	3	-	-	-	3
CO 2	3	3	3	3	3	-	2	3	-	3	2	3
CO 3	3	3	3	3	3	2	3	3	-	2	3	3
CO 4	3	3	2	2	-	-	-	3	3	3	3	3
CO 5	3	-	-	-	2	-	-	3	2	3	2	3

3/2/1: high/medium/low

Assessment Pattern

The End Semester Evaluation (ESE) will be conducted as an internal evaluation based on the product, the report and a viva- voce examination, conducted by a 3-member committee appointed by Head of the Department comprising HoD or a senior faculty member, academic coordinator for that program and project guide/coordinator. The Committee will be evaluating the level of completion and demonstration of functionality/specifications, presentation, oral examination, working knowledge and involvement.

The Continuous Internal Evaluation (CIE) is conducted by evaluating the progress of the mini project through minimum of TWO reviews. At the time of the 1st review, students are supposed to propose a new system/design/idea, after completing a thorough literature study of the existing systms under their chosen area. In the 2nd review students are expected to highlight the implementation details of the proposed solution. The review committee should assess the extent to which the implementation reflects the proposed design. A well coded, assembled and completely functional product is the expected output at this stage. The final CIE mark is the average of 1st and 2nd review marks.

A zeroth review may be conducted before the beginning of the project to give a chance for the students to present their area of interest or problem domain or conduct open brain storming sessions for innovative ideas. Zeroth review will not be a part of the CIE evaluation process.

Marks Distribution

Total Marks	CIE	ESE
150	75	75

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Marks awarded by Guide : 15 marks
Project Report : 10 marks
Evaluation by the Committee : 40 Marks

End Semester Examination Pattern: The following guidelines should be followed regarding award of marks.

(a) Demonstration : 50 Marks
(b) Project report : 10 Marks
(d) Viva voce : 15marks

Course Plan

In this course, each group consisting of three/four members is expected to design and develop a moderately complex software/hardware system with practical applications. This should be a working model. The basic concept of product design may be taken into consideration.

Students should identify a topic of interest in consultation with Faculty-in-charge of miniproject/Advisor. Review the literature and gather information pertaining to the chosen topic. State the objectives and develop a methodology to achieve the objectives. Carryout the design/fabrication or develop codes/programs to achieve the objectives. Demonstrate the novelty of the project through the results and outputs. The progress of the mini project is evaluated based on a minimum of two reviews**.**

The review committee may be constituted by the Head of the Department. A project report is required at the end of the semester. The product has to be demonstrated for its full design specifications. Innovative design concepts, reliability considerations, aesthetics/ergonomic aspects taken care of in the project shall be given due weight.

SEMESTER VI

PROGRAM ELECTIVE I

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT312	USER INTERFACE AND USER EXPERIENCE DESIGN	PEC	2	1	0	3

Preamble: User Interface and User Experience Design course is intended to deliver students the elementary concepts of User Interface Design, User Experience Design and their importance, thereby equipping them to develop great user interfaces which are appealing to users.

Prerequisite: nil

Course Outcome (CO): After completion of the course, the student will be able to

CO No.	Course outcomes (CO)	Bloom's Category Level
CO1	Outline the Basic Principles of Design and User Centered Design	Level 2: Understand
CO2	Infer the basic elements of User Experience Design	Level 2: Understand
CO3	Apply basic principles of Visual Design	Level 3: Apply
CO4	Discuss basic concepts in User Interface Design	Level 2: Understand
CO5	Develop Web and Mobile User Interface	Level 3: Apply

Mapping of Course Outcomes with Program Outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	1	1	1	-	-	-	-	-	2	2	3
CO2	3	1	1	1	-	-	-	-	-	2	2	2
CO3	3	3	3	1	-	-	-	-	-	2	1	2
CO4	3	1	2	1	-	-	-	-	-	2	1	2
CO5	3	3	3	1	-	-	-	-	-	3	2	3

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination (Marks)
	Test 1 (Marks)	Test 2 (Marks)	
Remember	5	5	10
Understand	30	30	60
Apply	15	15	30
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain the importance of seven stages of action in design.
2. Discuss Principles of design for understandability and usability

Course Outcome 2 (CO2):

1. Illustrate basic process components of UX.
2. Summarize basic process components of UX.

Course Outcome 3 (CO3):

1. Demonstrate the importance of typography with examples.
2. Summarize the basic process components of UX.

Course Outcome 4(CO4):

1. Explain the importance of a Good user interface design.
2. Differentiate two types of knowledge required for User Interface Design

Course Outcome 5 (CO5):

1. Create a web application for library with HTML and CSS.
2. List out any three mobile app designing tools.

Model Question Paper

Course Code: ITT312

Course Name: User Interface and User Experience Design

Max.Marks:100 **Duration: 3 Hours**

PART A

Answer all Questions. Each question carries 3 Marks

1. Define the terms understandability and usability in design.
2. Differentiate knowledge in the world and in the head in connection with design.
3. Explain User Experience.
4. Differentiate HCI and UX.
5. Differentiate Visual Design and fine arts.
6. Illustrate Iconography with an example.
7. Explain the importance of good user interface design.
8. Explain two types of knowledge needed for User Interface design.
9. What are the different design issues for websites?
10. Differentiate inline and block level elements.

PART B

Answer all questions. Each question carries 14 Marks

11. Illustrate the importance of seven stages of action as design aids with examples.

OR

12. Explain seven principles for transforming difficult tasks into simple one.

13. Illustrate funnel model for Agile UX with example.

OR

14. Explain 6D UX process.

15. Illustrate visual design process with example.

OR

16. Demonstrate the importance of any three visual design tools.

17. How to gather requirements for user interface designing?

OR

18. Demonstrate the importance of any three visual design tools.

19. Illustrate different ways for applying css with example.

OR

20. Differentiate stateful and stateless widgets in flutter.

Syllabus

Module 1: INTRODUCTION TO DESIGN (7 Hours)
Fundamentals of Design: Principles of design for understandability and usability, The Principle of mapping, How people do things: the seven stages of action, The seven stages of action as design aids, The trade of between knowledge in the world and in the head, Classification of everyday constraints, Applying affordance and constraints to everyday objects, The structure of tasks, The natural evolution of design, The complexity of design process **User Centered Design:** Seven Principles for transforming difficult tasks into simple one, Deliberately making the things difficult, Design and society, The design of everyday things
Module 2: FUNDAMENTALS OF USER EXPERIENCE DESIGN (7 Hours)
Basics of UX Design: The expanding concept of interaction, Definition of UX, UX Design, The components of UX, What UX is not, Kinds of interaction and UX, The basic process components of UX, UX Design techniques as life skills, Choosing UX Processes Methods and Techniques, The funnel model of Agile UX, Shifting paradigms in HCI and UX **Introduction To 6d:** 6D UX process – Discover, Design, Dream, Design, Develop, Deliver.
Module 3: VISUAL DESIGNING (7 Hours)
Introduction To Visual Design: The visual brain, Benefits to learning, The picture superiority effect, Visual design versus fine arts, The purpose of design, The role of the visual designer, A visual design process, The mindset of the visual designer, Build a graphic design

toolbox, Design with templates, Where to find visual inspiration, Know the technical terms, Use color with purpose, Establish a visual hierarchy, Tell stories with visuals, draft your verbal brand, Anatomy of typeface, color theory, understanding visual weight **VISUAL DESIGN TOOLS:** visual design tools – Photoshop, Illustrator, Creating layouts, iconography, digital color schemes, infographics, typography
Module 4: USER INTERFACE DESIGNING (6 Hours)
Basics Of Ui Development: Why the user interface matters?, The importance of good user interface design, Designing for users, Evaluation, How to gather requirements: Observing your users, Interviewing users, Questionnaire and surveys; Finding out about users and domain: Users-finding out who they are, User's needs, The domain; Describing user's work, Two types of knowledge needed for User Interface design, Design principles and design rules, Usability requirements, The Modern day view of usability, wireframing, prototyping **Wireframing And Prototyping Tools:** AdobeXD, Invision, AxureRP - overview
Module 5: UI DESIGNING FOR WEB AND MOBILE (8 Hours)
Designing for Web: Design principles for websites, Designing Websites, Designing Home pages and Interior pages, Design issues for web pages, Writing contents for web pages **Front End Development:** Front-end development technologies – HTML, Structure of HTML Page, Mandatory tags in html page (html, head, body), Heading tags (H1…H6), Tags and attributes (Class, Id, style etc.). Inline and block level elements, CSS, Different ways of applying CSS for elements, Responsive Web Designing, Bootstrap, Material Design, DOM, JQuery- animations **Mobile Front End Development:** Mobile App Designing tools- Sketch, Invisio, Adobe XD, Fluid; Mobile App Development- fundamentals, Android studio vs Flutter, Flutter framework- stateful and stateless widgets, Material icons, Basic app Development with Flutter

Textbooks

1. The Design of Everyday things- Donald A Norman, Currency and Doubleday, 2nd Edition
2. The UX Book-Rex Hartson and PardhaPyla, Morgan Kaufmann, 2nd Edition
3. Visual Design Solutions- Connie Malamed, Wiley, 1st Edition
4. User Interface Design and Evaluation - Debbie Stone,The Open University, 2nd Edition

Reference Books

1. Graphic Design For Everyone - Cath Caldwell, 2nd Edition
2. Adobe Photoshop, Illustrator, In-Design Basics - John Richards, 3rd Edition
3. Adobe XD classroom in a book - Brian Wood, 2nd Edition
4. How to build a web app in 4 stages -KarimAraoui
5. Axure RP Prototyping cookbook - John Henry Krahenbuhl

6. UI/UX Sketchbook for wireframing and prototyping - Amazon Digital Services LLC
7. Mobile UI/UX Sketchbook - (Independently published)
8. HTML and CSS: Design and Build Websites - Jon Duckett, Wiley
9. Flutter: for absolute beginners -PouyaHosseini
10. Seductive Interaction Design: Creating Playful, Fun, and Effective User Experiences - Stephen P Anderson,

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**INTRODUCTION TO DESIGN**	**7 Hours**
1.1	**Fundamentals of Design:** Principles of design for understandability and usability, The Principle of mapping, How people do things: the seven stages of action	2 Hours
1.2	The seven stages of action as design aids, The trade of between knowledge in the world and in the head, Classification of everyday constraints	2 Hours
1.3	, Applying affordance and constraints to everyday objects, The structure of tasks, The natural evolution of design, The complexity of design process	1 Hour
1.4	**User Centered Design:** Seven Principles for transforming difficult tasks into simple one, Deliberately making the things difficult, Design and society, The design of everyday things	2 Hours
2	**FUNDAMENTALS OF USER EXPERIENCE DESIGN**	**7 Hours**
2.1	**Basics Of UX Design:**The expanding concept of interaction, Definition of UX, UX Design, The components of UX	2 Hours
2.2	What UX is not, Kinds of interaction and UX, The basic process components of UX,	1 Hour
2.3	UX Design techniques as life skills, Choosing UX Processes Methods and Techniques, The funnel model of Agile UX, Shifting paradigms in HCI and UX	2 Hours
2.4	**Introduction To 6d:** 6D UX process – Discover, Design, Dream, Design, Develop, Deliver.	2 Hours
3	**VISUAL DESIGNING**	**7 Hours**
3.1	**Introduction To Visual Design:** The visual brain, Benefits to learning,	2 Hours

	The picture superiority effect	
3.2	Visual design versus fine arts, The purpose of design, The role of the visual designer, A visual design process, The mindset of the visual designer	1 Hour
3.3	, Build a graphic design toolbox, Design with templates, Where to find visual inspiration, Know the technical terms, Use color with purpose, Establish a visual hierarchy, Tell stories with visuals, draft your verbal brand, Anatomy of typeface, color theory, understanding visual weight	3 Hours
3.4	**VISUAL DESIGN TOOLS:** visual design tools – Photoshop, Illustrator, Creating layouts, iconography, digital color schemes, infographics, typography	1 Hour
4	**USER INTERFACE DESIGNING**	**6 Hours**
4.1	**Basics Of Ui Development:** Why the user interface matters?, The importance of good user interface design, Designing for users, Evaluation,	1 Hour
4.2	How to gather requirements: Observing your users, Interviewing users, Questionnaire and surveys; Finding out about users and domain: Users-finding out who they are, User's needs, The domain; Describing user's work,	2 Hours
4.3	Two types of knowledge needed for User Interface design, Design principles and design rules, Usability requirements, The Modern day view of usability, wireframing, prototyping	2 Hours
4.4	**Wireframing And Prototyping Tools:** AdobeXD, Invision, AxureRP - overview	1 Hour
5	**UI DESIGNING FOR WEB AND MOBILE**	**8 Hours**
5.1	**Designing for Web:** Design principles for websites, Designing Websites, Designing Home pages and Interior pages, Design issues for web pages, Writing contents for web pages	1 Hour
5.2	**Front End Development:**Front-end development technologies – HTML, Structure of HTML Page, Mandatory tags in html page (html, head, body), Heading tags (H1…H6), Tags and attributes (Class, Id, style etc.), Inline and block level elements,	2 Hours
5.3	CSS, Different ways of applying CSS for elements, Responsive Web Designing, Bootstrap, Material Design, DOM, JQuery- animations	2 Hours

5.4	**Mobile Front End Development:** Mobile App Designing tools- Sketch, Invisio, Adobe XD, Fluid; Mobile App Development- fundamentals,	1 Hour
5.5	Android studio vs Flutter, Flutter framework- stateful and stateless widgets, Material icons, Basic app Development with Flutter	2 Hours

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT322	COMPILER DESIGN	PEC	2	1	0	3

Preamble: This syllabus starts with an introduction about different phases of compilers. Each phase is discussed in detail in other modules. Second and third module contains the lexical analysis phase and automata for this phase. Parsing and different types of parsing techniques are being discussed over here. Different types of construction of parsing tables are included. Different syntax directed translation schemes are included in fourth module. Different code optimization techniques and code generation design issues are also included in the last module.

Prerequisite: ITT 307 Formal Languages and Automata theory

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome(CO)	Bloom's Category Level
CO 1	Explain the different phases of a compilers	Level 1: Remember
CO 2	Illustrate different automata, context free grammars for lexical analysis and parsing	Level 2: Understand
CO 3	Compare top-down parsers with bottom-up parsers	Level 3: Apply
CO 4	Construct different parsing tables for SLR, LALR ,LR, CLR	Level 3: Apply
CO 5	Illustrate the different syntax directed translation schemes , different code optimization techniques and code generation issues	Level 2: Understand

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	2	2	-	-	-	-	-	-	-	1
CO 2	3	2	2	2	1	-	-	-	-	-	-	1
CO 3	2	3	2	3	-	-	-	-	-	-	-	1
CO 4	3	3	3	3	3	-	-	-	-	-	-	1
CO 5	2	2	2	3	3	-	-	-	-	-	-	1

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test1 (Marks)	Test2 (Marks)	
Remember	10	10	20
Understand	25	25	50
Apply	15	15	30
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. State the different phases of a compiler.

2. List the different Finite Automata for token recognition, Lexical Analysis.

3. Define the different compiler writing tools.

Course Outcome 2 (CO2)

1. Demonstrate different context free grammars and derivation trees for ambiguous grammar.

2. Give example for recursive descent parsing.

3. Construct parsing table for LL(1) Grammar.

Course Outcome 3(CO3):

1. Demonstrate first and follow.

2. Construct parsing table for SLR.

3. Construct parsing table for LALR.

Course Outcome 4 (CO4):

1. Demonstrate the syntax directed translation for different statements.

2. Give example for generating physical address for accessing data segment.

3. Describe the bottom-up evaluation and top-down evaluation of inherited attributes.

Course Outcome 5 (CO5):

1. Illustrate the three-address code generation in different formats.

2. Demonstrate different code optimization techniques.

3. Demonstrate the design of code generator.

Model Question paper

Course Code: ITT322
Course Name: COMPILER DESIGN

Max. Marks: 100 Duration: 3 Hours

PART A

(Answer all questions. Each carries 3 marks)

1. Define ambiguous grammar. Give an example.
2. Write a note on derivation tree with an example.
3. Discuss about input buffering.
4. Define Lexemes, token and symbol table.
5. Write a note on operator precedence parsing.
6. What is shift reduce parsing with an example
7. Define synthesized and inherited attribute
8. Write a note on type checking
9. What are the methods used for elimination of ambiguity and non-determinism?
10. Write a note on
    - L attributed definition
    - S attributed definition

PART B

Answer all questions. Each carries 14 marks

11. Write regular expression for the following language (14 marks)
    a. Set of all strings that contains 11 at the beginning and 00 at the end where$\sum$={0,1} and draw a corresponding E-NFA
    b. Set of all strings that contains 3 consecutive a's where$\sum$={a,b} and draw a

corresponding E-NFA

OR

12. Explain the different phases of compiler with neat diagram (14marks)

13. Design CFG for the following language and draw corresponding parse tree (14 marks)

a. Design a CFG for of $a^n b^n$

b. Design a CFG for equal number of zeros and equal number of ones

OR

14 a)Define LR Parsing. Construct LR(0) items for the below grammar. (7 marks)

S->AA

A->aAb

b)Differentiate between LR(0) parsing and SLR Parsing with an example (7marks)

15 . a.Write an algorithm for recursive descent parser (7 marks)

b.Write an algorithm for FIRST and FOLLOW in parser (7 marks)

OR

16. Explain LALR parsing with an example (14 marks)

17. Describe about syntax directed definition with examples (14 marks)

OR

18. Discuss about bottom up evaluation of synthesized attributes (14 marks)

19 a. Discuss about 1. Three address code 2.Quadruples 3.Triples with an example (7 marks)

b. Discuss about various storage allocation strategies (7 marks)

OR

20 a. Discuss about any three code optimization methods (7marks)

b. Write a note on various issues of code generation (7 marks)

Syllabus

Module 1: Introduction to compilers (6 Hours)
Introduction to compilers – Analysis of the source program, Phases of a compiler, Grouping of phases, compiler writing tools – bootstrapping **Lexical Analysis:** The role of Lexical Analyser, Input Buffering, Review of Finite Automata.
Module 2: Syntax Analysis (6 Hours)
Syntax Analysis: Review of Context-Free Grammars – Derivation trees and Parse Trees, Ambiguity. **Top-Down Parsing:** Recursive Descent parsing, Predictive parsing, LL(1) Grammars.
Module 3: Bottom-Up Parsing (9 Hours)
Bottom-Up Parsing: Shift Reduce parsing – Operator precedence parsing **,**LR parsing – Constructing SLR parsing tables, Constructing, Canonical LR parsing tables and Constructing LALR parsing tables.
Module 4: Syntax directed translation (6 Hours)
Syntax directed translation: Syntax directed definitions, Bottom- up evaluation of S-attributed definitions, L- attributed definitions, Top-down translation, Bottom-up evaluation of inherited attributes.
Module 5: Intermediate Code Generation (ICG) (8 Hours)
Intermediate Code Generation (ICG): Intermediate languages – Graphical representations, Three-Address code, Quadruples, Triples. **Code Optimization:** Principal sources of optimization, Optimization of Basic blocks **Code generation:** Issues in the design of a code generator, A simple code generator.

Text Books

1. Aho A. Ravi Sethi and D Ullman. Compilers – Principles Techniques and Tools, Addison Wesley, 2006.

2. D. M.Dhamdhare, System Programming and Operating Systems,Tata McGraw Hill & Company, 1996.

Reference Books

1. Kenneth C. Louden, Compiler Construction – Principles and Practice, Cengage Learning Indian Edition, 2006.

2. Tremblay and Sorenson, The Theory and Practice of Compiler Writing, Tata McGraw Hill & Company,1984.

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction to compilers**	**6 Hours**
1.1	Analysis of the source program	1 Hour
1.2	Phases of a compiler	1 Hour
1.3	Grouping of phases, compiler writing tools ,bootstrapping	1 Hour
1.4	**Lexical Analysis:** The role of Lexical Analyser, Input Buffering	1 Hour
1.5	Review of Finite Automata	2 Hours
2	**Syntax Analysis**	**6 Hours**
2.1	Review of Context-Free Grammars	1 Hour
2.2	Derivation trees and Parse Trees, Ambiguity.	1 Hour
2.3	Recursive Descent parsing	2 Hours
2.4	Construction of LL(1) parsing table	2 Hours
3	**Bottom-Up Parsing**	**9 Hours**
3.1	Shift Reduce parsing	1 Hour
3.2	Operator precedence parsing	2 Hours
3.3	SLR parsing tables	2 Hours
3.4	Constructing LALR parsing tables	2 Hours
3.5	Canonical LR parsing tables	2 Hours
4	**Syntax directed translation**	**6 Hours**
4.1	Syntax directed definitions	1 Hour
4.2	Bottom- up evaluation of S- attributed definitions	2 Hours
4.3	L- attributed definitions	1 Hour
	Top-down translation	1 Hour
	Bottom-up evaluation of inherited attributes.	1 Hour
5	**Intermediate Code Generation (ICG)**	**8 Hours**
5.1	Intermediate languages – Graphical representations, Three-Address code, Quadruples, Triples.	2 Hours
5.2	**Code Optimization:** Principal sources of optimization	2 Hours
5.3	Optimization of Basic blocks	2 Hours
5.4	**Code generation:** Issues in the design of a code generator	1 Hour
5.5	A simple code generator	1 Hour

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT332	SOFT COMPUTING	PEC	2	1	0	3

Preamble: The objective of the course is to introduce the basic concepts of soft computing techniques such as Artificial Neural Networks, Fuzzy Logic, Genetic Algorithm and Hybrid Systems.

Prerequisite: Nil

Course Outcomes: After the completion of the course, the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Explain various soft computing techniques and their roles in building intelligent machines	Level 2: Understand
CO 2	Discuss Artificial Neural Network Architectures and different Learning Methods	Level 2: Understand
CO 3	Apply fuzzy logic and reasoning to handle uncertainty and solve engineering problems	Level 3: Apply
CO 4	Apply genetic algorithm to solve optimization problems	Level 3: Apply
CO 5	Explain the concepts of hybrid systems	Level 2: Understand

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	2	1	-	-	-	-	-	-	-	-	-	2
CO 2	3	3	-	1	-	-	-	-	-	-	-	2
CO 3	3	3	3	3	-	-	-	-	-	-	-	2
CO 4	3	3	3	3	-	-	-	-	-	-	-	2
CO 5	2	1	-	1	-	-	-	-	-	-	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test1 (Marks)	Test1 (Marks)	
Remember	20	10	30
Understand	20	20	40
Apply	10	20	30
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 10 marks
Continuous Assessment Test (2 numbers)	: 25 marks
Assignment/Quiz/Course project	: 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Describe various soft computing techniques.
2. List applications of Neural Networks.
3. Soft computing techniques give best solution to complex problems. Justify.

Course Outcome 2 (CO2):

1. What is unsupervised learning and how is it different from reinforced learning.

2. How does learning takes place in supervised learning.

3. Draw the architecture of back propagation algorithm.

Course Outcome 3(CO3):

1. With suitable example, explain how membership assignment is performed using intuition.

2. Design computer software to perform image processing to locate objects within a scene. The two fuzzy sets representing a plane and a train image are

$$\text{Plane}: \left\{ \frac{0.2}{train} + \frac{0.5}{bike} + \frac{0.3}{boat} + \frac{0.8}{plane} + \frac{0.1}{house} \right\}$$

$$\text{Train}: \left\{ \frac{1}{train} + \frac{0.2}{bike} + \frac{0.4}{boat} + \frac{0.5}{plane} + \frac{0.2}{house} \right\}$$

Find the following

(a) Plane □ Train

(b) Plain □ Train

(C) $\overline{Plane}$

(d) $\overline{Train}$

(e) Plane□ Train

(f) $\overline{\text{Plane} \cup \text{Train}}$

(g) $\overline{\text{Plane} \cap \text{Train}}$

(h) $\overline{\text{Plane} \cup \overline{\text{Plane}}}$

(i) Plain □ $\overline{Plane}$

(j) Train □ $\overline{Train}$

Course Outcome 4 (CO4):

1. Determine the maximum of a function $f(x)= x^2$ using genetic algorithm.

2. With a neat flowchart, explain the operation of a simple genetic algorithm.

Course Outcome 5 (CO5):

1. Describe Neuro Genetic hybrid systems.

2. Mention the characteristics and properties of Neuro-Fuzzy hybrid systems

Model Question paper

Course Code: ITT 332

Course Name: Soft Computing

Max.Marks:100 **Duration: 3 Hour**

Part A

Answer all questions. Each question carries 3 marks.

1. Discuss the back propagation process in a neural network.
2. How is fuzzy relation converted into a crisp relation using lamda-cut process?
3. Differentiate convex and nonconvex fuzzy set
4. What is ANFIS?
5. Differentiate hard computing and soft computing
6. What is the significance of weight in an Artificial Neural Network?
7. Define Fuzzy Equivalence Relation.
8. Compare Tuning and learning problems
9. What are the advantages and limitations of Genetic Algorithm?
10. List various encoding techniques used in genetic algorithm.

Part B

Answer all questions. Each question carries 14 marks. (5 * 14 = 70 Marks)

11. (a) State the basic components of soft computing (6 marks)

(b) What are the different applications of Soft Computing? (8 marks)

OR

12. (a) What are the characteristics of hard computing and soft computing? (6 marks)

(b) Describe various soft computing techniques (8 marks)

13. (a)Explain Exclusive OR problem. How it is solved with two layer perceptrons (8 marks)

(b) Calculate the output y of a three-input neuron with bias. The input feature vector is (x1, x2,x3)= (0.3,0.5,0.6) and weight values are [w1,w2,w3, b] = [0.1, 0.3, -0.2, 0.35]. Use (i) binary sigmoidal and (ii) bipolar sigmoidal activation functions (6 marks)

OR

14. (a) (i) Construct a feed forward network with five input nodes, three hidden nodes and four output nodes.

(ii) Construct a recurrent network with four input nodes, three hidden nodes and two output nodes that has feedback links from the hidden layer to the input layer.

(8 marks)

(b) Compare Supervised and Unsupervised Learning Methods. (6 marks)

15. (a) Using the inference approach, find the membership values for the fuzzy triangular shapes(i) isosceles triangle,(ii) equilateral triangle,(iii) right angle triangle,(iv) isosceles and right angle triangle (v) other triangles for a triangle with angles 45^0,55^0,80^0 (10 marks)

(b) What are the different features of membership functions? (4 marks)

OR

16. (a) Explain different Defuzzification methods (8 marks)

(b) Describe Max-min composition and Max-product composition of Classical Relations

(6 marks)

17. (a) Define the following Aggregation of Fuzzy Rules

(i) Conjunctive system of rules (ii) Disjunctive system of rules (6 marks)

(b) Explain four modes of Fuzzy Approximate Reasoning (8 marks)

OR

18. (a) Compare Mamdani Fuzzy Interface System and Takagi-Sugeno Fuzzy Model

(8 marks)

(b) What is meant by compound rule? List the different methods used for decomposition of compound linguistic rules into simple canonical rules. (6 marks)

19. (a) Briefly explain the selection operation in genetic algorithm. (5 marks)

(b) Compare and contrast cooperative Neuro- fuzzy systems and hybrid Neuro-fuzzy systems. (9 marks)

OR

20. (a) Explain Two-Point Crossover. In a Genetic algorithm, suppose that two potential parents are given by

1	1	0	0	1	1	0	1	1	1
0	0	1	1	1	0	1	0	0	1

Assuming the numbering goes from left to right and that $\square_1=4$ and $\square_2=8$, show result of two point crossover

(6 marks)

(b) Describe Neuro Genetic Hybrid Systems (8 marks)

Syllabus

Module 1: Introduction to Soft Computing (5 Hours)
Evolution of Computing-From Conventional Artificial Intelligence to Computational Intelligence, Characteristics of Hard Computing and Soft Computing, Soft Computing Constituents, Applications of Soft Computing
Module 2: Artificial Neural Networks (7 Hours)
Biological Neuron, Artificial Neural Network Architectures: Single-Layer Feed Forward Network, Multi-Layer Feed Forward Network and Recurrent Network, Learning Methods: Supervised, Unsupervised and Reinforced Learning
Module 3: Fuzzy Logic (8 Hours)
Fuzzy Sets, Operations on Fuzzy Sets, Fuzzy Relations, Operations on Fuzzy Relations, Fuzzy Membership Functions, Fuzzification, Methods of Membership Value Assignments, Defuzzification Methods
Module 4: Fuzzy System (6 Hours)
Fuzzy Rules: Formation, Decomposition and Aggregation, Fuzzy Reasoning, Fuzzy Inference System: Mamdani Fuzzy System and Sugeno Fuzzy System
Module 5: Genetic Algorithm and Hybrid Systems (9 Hours)
Genetic Algorithm: Basic Version of Genetic Algorithm, Encoding Methods, Operators in Genetic Algorithm: Selection, Crossover and Mutation Hybrid Systems: Basic Concept, Neuro-Fuzzy Hybrid System, Neuro-Genetic Hybrid System and Fuzzy-Genetic Hybrid System

Text Books

1. S.N.Sivanandam , S.N.Deepa, Principles of Soft Computing, Wiley India Pvt. Ltd., 2nd Edition, 2011.
2. S.Rajasekaran, G.A.Vijayalakshmi Pai, Neural Networks, Fuzzy Logic and Genetic Algorithm, Synthesis and Applications , PHI Learning Pvt. Ltd., 2017.

Reference Books

1. James A. Freeman and David M. Skapura, ―Neural Networks Algorithms, Applications, and Programming Techniques, Addison Wesley, 2003.
2. Jacek M. Zurada, Introduction to Artificial Neural Systems, PWS Publishers, 1992.
3. George J. Klir and Bo Yuan, ―Fuzzy Sets and Fuzzy Logic-Theory and Applications, Prentice Hall, 1996.
4. Mitchell Melanie, An Introduction to Genetic Algorithm, Prentice Hall, 1998.
5. David E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Addison Wesley,1997.
6. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, ―Neuro-Fuzzy and Soft Computing, Prentice-Hall of India, 2002.

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction to Soft Computing**	**5 Hours**
1.1	Evolution of Computing-From Conventional Artificial Intelligence to Computational Intelligence	2 Hours
1.2	Characteristics of Hard Computing and Soft Computing,	1 Hour
1.3	Soft Computing Constituents	1 Hour
1.4	Applications of Soft Computing	1 Hour
2	**Artificial Neural Networks**	**7 Hours**
2.1	Biological Neuron and Artificial Neural Network Concepts	1 Hour
2.2	Single-Layer and Multi-Layer Feed Forward Networks	2 Hours
2.3	Recurrent Network	1 Hour
2.4	Supervised Learning	1 Hour
2.5	Unsupervised Learning	1 Hour
2.6	Reinforced Learning	1 Hour
3	**Fuzzy Logic**	**8 Hours**
3.1	Fuzzy Sets and Operations on Fuzzy Sets	2 Hours
3.2	Fuzzy Relations and Operations on Fuzzy Relations	2 Hours
3.3	Fuzzy Membership Functions	2 Hours
3.4	Fuzzification and Methods of Membership Value Assignments	1 Hour
3.5	Defuzzification Methods	1 Hour
4	**Fuzzy System**	**6 Hours**
4.1	Fuzzy Rules: Formation, Decomposition and Aggregation	2 Hours
4.2	Fuzzy Reasoning	2 Hours
4.3	Fuzzy Inference System: Mamdani and Sugeno Fuzzy Systems	2 Hours
5	**Genetic Algorithm and Hybrid Systems**	**9 Hours**
5.1	Basic Version of Genetic Algorithm	1 Hour

5.2	Encoding Methods	1 Hour
5.3	Operators in Genetic Algorithm: Selection, Crossover and Mutation	3 Hours
5.4	Basic Concept of Hybrid Systems	1 Hour
5.5	Neuro-Fuzzy Hybrid System	1 Hour
5.6	Neuro-Genetic Hybrid System	1 Hour
5.7	Fuzzy-Genetic Hybrid System	1 Hour

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT342	MICROPROCESSORS	PEC	2	1	0	3

Preamble: This course is intended to deliver students the concepts of Microprocessors and Micro-controllers. It also helps them to learn how to write an 8086 program assembly language. Introduction to Interfacing of micro-processors, its use and applications are also covered in the syllabus

Prerequisite:ITT204 Computer Organization

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Explain different modes of operations of a typical microprocessor and microcontroller.	Level 2: Understand
CO 2	Develop assembly language programs for problem solving using software interrupts and various assembler directives.	Level 3: Apply
CO 3	Illustrate how to handle 8086 interrupts	Level2: Understand
CO 4	Identify interfacing of various I/O devices to the microprocessor through assembly language programming	Level 2: Understand
CO 5	Develop assembly language programs using 8051 microcontrollers.	Level 3: Apply

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	-	-	-	-	-	-	-	-	-	-	1
CO 2	3	3	3	-	2	-	-	-	-	-	-	2
CO 3	3	2	-	2	-	-	-	-	-	-	-	1
CO 4	3	3	3	2	3	-	-	-	3	2	3	3
CO 5	3	3	3	-	-	-	-	-	-	-	-	3

3/2/1:high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test 1(Marks)	Test2(Marks)	
Remember	10	10	20
Understand	25	25	50
Apply	15	15	30
Analyse			

Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. State the functionality of a finite state machine and the microprocessor.

2. List the microprocessors launched by various company.

3. Define the functionality of a microprocessor.

Course Outcome 2 (CO2)

1.Demonstrate the segment override instructions in x86

2. List the registers present in x86 microprocessors.

3. Define the functionality of a stack and stack pointer.

Course Outcome 3(CO3):

1. Illustrate that how to handle interrupts in x86 architecture.

2. Show the program for accessing a maskable interrupt in x86.

3. Discuss Interfacing with 8259.

Course Outcome 4 (CO4):

1. Explain the Interfacing and mode of operation of 8257 with 8086.

2.Outline theInterfacing 8253 with 8086.

3.Write a program to do LED display using 8279 with 8086.

Course Outcome 5 (CO5):

1.List criteria for Selecting a microcontroller for Applications.

2.Draw and explain 8051 Architecture

3.Explain Memory and I/O addressing with 8051

Model Question paper

Course Code: ITT342
Course Name: MICROPROCESSORS

Max. Marks: 100 **Duration: 3 Hours**

PART A

(Answer all questions. Each question carries 3 Marks)

1.Compare microprocessor and microcomputer.

2.List the instructions which are used for memory operation in 8088 microprocessors.

3.Explain the purpose of the following signals in 8086.

(i) READY (ii) HOLD

4.Write a program to do data conversions from HEX to ASCII in 8086 assembly code.

5.Explain MOV, AAA, HLT instructions of 8086 Micro-processor.

6. Illustrate with an example how arrays are used in 8086.

7.Describe different modes of operation of the following peripheral ICs:

i) 8255 ii) 8257

8.Write the Control Word Format in 8255.

9.Write any three applications of microcontrollers.

10. Write short notes on memory and IO addressing of 8051 microcontrollers.

PART B

(Answer all questions. Each full question carries 14 marks)(5*14=70)

11.a) Draw the architectural diagram of 8086 microprocessor and explain. (8 marks)

b) Draw the pin diagram of 8086 micro-processor and explain the function of each pin. (6 marks)

OR

12. a) Compare 8086 and 8088 processors. (7 marks)

b) Explain 8086 Addressing mode (7 marks)

13.a) Write an 8086 program to perform linear search (8 marks)

b) An array of 10 numbers is stored in the internal data RAM starting from location 30 H. Write an assembly language program to sort the array in ascending order starting from location 40 H. (7 marks)

OR

14.a) Write an 8086 based assembly language program to perform addition of two 2x2 matrices. (7 marks)

b) Write an assembly program to add N numbers. (5 marks)

15.a) Explain multiple Interrupt handling. (6 marks)

b) Explain the architecture of programmable interrupt controller 8259. (8 marks)

OR

16. a) What are different types of Interrupts? What is Interrupt service Routine? (6 marks)

b) Draw the memory map and briefly explain the memory organization for 128-byte internal RAM of 8086 micro-controller. (8 marks)

17.a) Explain 8255 Interfacing (7 marks)

b) Explain modes of 8257 (7 marks)

OR

18. a). Explain Keyboard Display controller 8279. (7 marks)

b) Give the advantage of using 8279 for keyboard/display interface? What are scan lines used for? Explain (i) Encoded Scan Mode and (ii) Decoded scan mode. (7 marks)

19.(a) List out criteria for selecting a microcontroller for Applications. (6 marks)

(b)Explain the interfacing of 8253 Timer using 8051 micro-controllers. (8 marks)

OR

20.(a) Explain the functions of ports in 8051 micro-controllers. How can P1 be used as both output and input port? (6 marks)

b) Discuss the addressing modes of 8051 Instruction set. (8 marks)

Syllabus

Module 1:Introduction to Microprocessors(6 Hours)
Evolution of microprocessors, 8086 Microprocessor – Architecture, Memory organisation, Minimum and maximum mode of operation, Minimum mode Timing Diagram. Comparison of 8086 and 8088, Pentium series -introduction to 8087 math coprocessors.8086 Addressing Modes.
Module 2: Instructions and Programming(8 Hours)
8086 Instruction set and Assembler Directives - Assembly Language Programming with Subroutines, Macros, Passing Parameters, Use of stack. Linking and Relocation - Stacks - Procedures – Timing and control unit, op-code fetch machine cycle, memory read/write machine cycles, I/O read/write machine cycles.
Module 3: Interrupt handling(6 Hours)
Interrupts:- Types of Interrupts and Interrupt Service Routine. Handling Interrupts in 8086, Interrupt programming. Handling multiple interrupts Basic Peripherals and their Interfacing with 8086 – Programmable Interrupt Controller - 8259 - Architecture.
Module 4: Interfacing(8 Hours)
Interfacing Memory, I/O, 8255 - Detailed study - Architecture, Control word format and modes of operation, Architecture and modes of operation of 8279 and 8257, programmable counter/interval timer (8253 and 8254)
Module 5: Microcontroller(7 Hours)
Microcontrollers - Types of Microcontrollers - Criteria for selecting a microcontroller - Example Applications. Characteristics and Resources of a microcontroller. Organization and design of these resources in a typical microcontroller - 8051. 8051 Architecture, Register Organization, Memory and I/O addressing, Interrupts and Stack.

Text Books

1.Muhammad Ali Mazidi,et.al,, The 8051 Microcontroller and Embedded Systems: Using Assembly and C, Pearson Education, 2007

2. Barry B. Brey, The Intel Microprocessors – Architecture, Programming and Interfacing, Eighth Edition, Pearson Education, 2015

3. Douglas V. Hall, SSSP Rao, Microprocessors and Interfacing, Third Edition, McGrawHill Education, 2012.

Reference Books

1.Bhurchandi and Ray, Advanced Microprocessors and Peripherals, Third Edition McGraw Hill, 2012

2. Raj Kamal, Microcontrollers: Architecture, Programming, Interfacing and System Design, Pearson Education, 2011.

3. A. NagoorKani, Microprocessors and Microcontrollers, Second Edition, Tata McGraw Hill, 2012.

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction to Microprocessors6 Hours**	
1.1	Evolution of microprocessors	1Hour
1.2	8086 Microprocessor – Architecture	1Hour
1.3	Memory organisation, Minimum and maximum mode of operation,	1Hour
1.4	Minimum mode Timing Diagram.	1(T)Hour
1.5	8086 Addressing Modes	1Hour
1.6	Comparison of 8086 , 8088 and Pentium series	1(T)Hour
2	**Instructions and Programming8 Hours**	
2.1	8086 Instruction set and Assembler Directives	2Hours
2.2	Assembly Language Programming with Subroutines, Macros, Passing Parameters	1Hour
2.3	Linking and Relocation - Stacks - Procedures	1Hour
2.4	Timing and control unit	1Hour
2.5	Op-code fetch machine cycle, memory read/write machine cycles, I/O read/write machine cycles,	2Hours
2.6	8086 Programming	1(T)Hour
3	**Interrupt handling6 Hours**	
3.1	Interrupts - Types of Interrupts and Interrupt Service Routine.	1Hour
3.2	Handling Interrupts in 8086,	1Hour
3.3	Handling multiple interrupts	1Hour

3.4	Basic Peripherals and their Interfacing with 8086	1Hour
3.5	Programmable Interrupt Controller - 8259 - Architecture.	1Hour
3.6	Interrupt programming.	1(T)Hour
4	**Interfacing8Hours**	
4.1	Interfacing Memory, I/O,	1Hour
4.2	8255 - Detailed study - Architecture	1Hour
4.3	Control word format and modes of operation	1Hour
4.4	Architecture and modes of operation of 8279	1Hour
4.5	Architecture and modes of operation of 8257	1Hour
4.6	Programmable counter/interval timer (8253 and 8254)	1Hour
4.7	Interfacing using Simulator	2(T)Hours
5	**Microcontroller7 Hours**	
5.1	Microcontrollers - Types of Microcontrollers - Criteria for selecting a microcontroller - Example Applications.	1Hour
5.2	Characteristics and Resources of a microcontroller. Organization and design of these resources in a typical microcontroller -8051	1 Hour
5.3	8051 Architecture, Register Organization,	2Hours
5.4	Memory and I/O addressing,	1Hour
5.5	Interrupts and Stack.	1Hour
5.6	Interfacing with Timer using simulator	1(T)Hour

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT352	DISTRIBUTED SYSTEMS	PEC	2	1	0	3

Preamble: This course covers introductory concepts in the design and implementation of distributed systems. The course aims to cover all the fundamental and applied techniques in distributed systems ranging from various distributed architectural styles to consistency, replication and fault tolerance in distributed systems. It also covers design of computer clusters for scalable parallel computing.

Prerequisites:

- ITT201 Data Structures
- ITT305 Data Communication and Networking

Course Outcomes: After the completion of the course the student will be able to

CO No	Course Outcomes (CO)	Bloom's Category Level
CO 1	Discuss the characteristics and design goals of basic distributed systems.	Level 2: Understand
CO 2	Apply knowledge of distributed systems techniques and methodologies.	Level 3: Apply
CO 3	Explain the design and development of distributed systems and distributed systems applications.	Level 3: Understand
CO 4	Use the application of fundamental computer science methods and algorithms in the development of distributed systems and distributed systems applications.	Level 3: Apply
CO 5	Interpret the design and testing of a large software system, and to be able to communicate that design to others.	Level 3: Apply

Mapping of course outcomes with program outcomes:

POs / Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	-	-	-	-	-	-	-	-	-	2
CO 2	3	3	-	1	-	-	-	-	-	-	-	2
CO 3	3	3	2	1	-	-	-	-	-	-	-	2
CO 4	3	3	3	1	-	-	-	-	-	-	-	2
CO 5	3	3	3	3	3	-	-	-	-	3	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test 1(Marks)	Test 2(Marks)	
Remember	10	10	10
Understand	20	20	40
Apply	20	20	50
Analyse			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Specify the characteristics of a distributed system.
2. What are the design objectives of computer clusters?

Course Outcome 2 (CO2):

1. Identify the major design issues of clustered and massively parallel processor system.
2. Differentiate and exemplify compact versus slack clusters.

Course Outcome 3(CO3):

1. Discuss about InfiniBand system fabric built in a typical high-performance computer cluster.
2. Give the organization of an internet search engine and explain the layers associated with it.

Course Outcome 4 (CO4):

1. With an example, explain how bully algorithm works.
2. Explain in detail checkpointing and recovery techniques used cluster systems?

Course Outcome 5 (CO5):

1. Specify the operate-repair cycle of a computer system.
2. Explain in detail how gossip-based publish-subscribe system handles subscriber queries.

Model Question paper

Course Code: ITT352
Course Name: Distributed Systems

Max.Marks:100 **Duration: 3 Hours**

PART A

(Answer all Questions. Each question carries 3 Marks)

1 What are the different forms of transparency in a distributed system?

2 What are the important role of wrappers?

3 Differentiate stateless and stateful serv(ers.

4 A major drawback of user level threads comes from deploying the many-to-one threading model. Explain.

5 Mention the properties of a true identifier.

6 What is the alternative technique used, when GPS is not an option?

7 Clarify the difference between consistency and coherence

8 What is the role of orphan process?

9 List the parts of a job management system.

10 Explain gang scheduling.

PART B

(Answer all questions. Each full question carries 14 marks) (5*14=70)

11.a. Illustrate a layered communication protocol stack?

OR

12.a With a neat diagram, explain the basic NFS architecture for unix system

13.a What is code migration? List the reasons for migrating code.

b. Explain how RPC achieves its transparency?

OR

14. What is virtualisation? Explain the different types of virtualisation based on the interfaces of a computer system.

15.a. Flat names are good for machines, but are generally not very convenient for humans to use. Explain the alternative human friendly way of naming system and the way these names are resolved to address.

b. How clock synchronisation happens in a wireless network?

OR

16.a. Explain attribute-based naming scheme.

b. With an example, explain how bully algorithm works.

17.a. Explain the three different types of replicas of content replication and placement

b. What is a dependable system? Mention the requirements of a distributed system for dependable.

OR

18.a. How client-centric consistency is implemented?

b. Give solution to reach reliable group communication.

19.a. Draw and explain the architecture of a computer cluster built with commodity hardware, software, middleware and network components supporting HA and SSI.

OR

20.a Explain the design principals of SSI.

Syllabus

Module 1: Introduction and Architectural style (5 Hours)
Introduction and Architectures – Characteristics – Design goals – Types of distributed systems - High performance distributed computing – Distributed information systems – Pervasive systems – Architectural styles – Middleware organization – System architecture – Centralized – Decentralized – Hybrid organizations - The Network File System.
Module 2: Processes & Communication (8 Hours)
Processes & Communication – Threads in distributed systems – Virtualization – Clients – Servers – code migration – Communication – layered protocols and types of communication – remote procedure call – Basic RPC operation – Parameter passing – RPC-based application support – Variations on RPC – Message-oriented communication – Simple transient messaging with sockets – Advanced transient messaging – Message-oriented persistent communication – multicast communication.
Module 3: Naming & Coordination (7 Hours)
Naming & Coordination – Names, identifiers, and addresses – Flat naming – Structured naming - Attribute-based naming – Coordination - Clock synchronization Logical clocks – Mutual exclusion - Election algorithms - Location systems
Module 4: Consistency, replication, and Fault tolerance (9 Hours)
Consistency, replication, and Fault tolerance – Reasons for replication - Replication as scaling technique - Data-centric consistency models - Client-centric consistency models - Replica management – Consistency protocols - Example: Caching and replication in the Web - Fault tolerance – Basic concepts - Failure models – Failure masking by redundancy - Process resilience – Resilience by process groups – Failure masking and replication – Consensus in faulty systems with crash failures – Example: Paxos – Consensus in faulty systems with arbitrary failures – limitations on realizing fault tolerance – Failure detection.
Module 5: Computer Clusters for Scalable Parallel Computing (6 Hours)
Computer Clusters for Scalable Parallel Computing – Clustering for Massive Parallelism – Computer Clusters and MPP Architectures – Design Principles of Computer Clusters – Cluster Job and Resource Management.

Text Books

1. Tanenbaum, Andrew S, Van. Steen, Maarten, " Distributed systems Principles and paradigms", Third Edition , Pearson, 2017.

2. Kai Hwang, Geoffrey C. Fox, Jack J. Dongarra , "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things.", Morgan Kaufman, 2012.

References

1. Distributed Systems: Concepts and Design, G. Coulouris, Jean Dollimore and Tim Kindberg, Addison Wesley, 4^{th} Edition
2. Principles of Computer System Design. Jerome Saltzer and M. Frans Kaashoek, Morgan Kaufmann.
3. Large-scale Incremental Processing Using Distributed Transactions and Notifications, Proceedings of the 9^{th} USENIX Symposium on Operating Systems Design and Implementation, 2010
4. The Google File System, Proceedings of the 19^{th} ACM Symposium on Operating Systems Principles, 2003
5. Vijay Garg, Elements of Distributed Computing, Wiley, 2002.
6. Nancy A. Lynch, Distributed Algorithms, Morgan Kaufmann, 1996.
7. Kurose and Ross, Computer Networking: A top-down approach, Pearson, 2013.

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction and Architectural style**	**5 Hours**
1.1	Introduction and Types of distributed systems	2 Hours
1.2	Architectural styles	2 Hours
1.3	Middleware organization - System architecture	1 Hour
2	**Processes & Communication**	**8 Hours**
2.1	Threads in distributed systems and Virtuliazation	2 Hours
2.2	Clients – Servers - code migration	2 Hours
2.3	Remote procedure call	3 Hours
2.4	Message-oriented communication	1 Hour
3	**Naming & Coordination**	**7 Hours**
3.1	Naming types	1 Hour
3.2	Clock synchronization Logical clock	2 Hours
3.3	Election algorithms	3 Hours
3.4	Location systems	1 Hour

4	**Consistency, replication, and Fault tolerance**	**9 Hours**
4.1	Replication and consistency models	3 Hours
4.2	Replica management - Consistency protocols	3 Hours
4.3	Failure models and Process resilience	3 Hours
5	**Computer Clusters for Scalable Parallel Computing**	**6 Hours**
5.1	Clustering for Massive Parallelism, MPP architectures	2 Hours
5.2	Design Principles of Computer Clusters	2 Hours
5.3	Cluster Job and Resource Management .	2 Hours

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT362	DIGITAL IMAGE PROCESSING	PEC	2	1	0	3

Preamble: This course is intended to make the students capable of Understanding the fundamental concepts and applications of digital image Processing and methods for image segmentation and compression. They should also be able to perform various basic operations such as transforms, morphological operations and filters in digital image processing

Prerequisite: Basic understanding on signals and systems

Course Outcomes: After the completion of the course the student will be able to

CO No	Course outcomes (CO)	Bloom's Category Level
CO 1	Demonstrate the fundamentals of image processing such as representation of digital images, pixel relationships and representation of color images.	Level 2: Understand
CO 2	Apply various image transforms and different image compression techniques.	Level 3: Apply
CO 3	Summarize various methods for digital image enhancement in spatial domain.	Level 2: Understand
CO 4	Explain various methods for digital image enhancement in frequency domain.	Level 2: Understand
CO 5	Discuss the various methods for image segmentation and morphological operations in digital image processing.	Level 2: Understand

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3	3	2	2	-	-	-	-	-	-	2
CO 2	3	2	2	1	2	-	-	-	-	-	-	2
CO 3	3	2	2	2	-	-	-	-	-	-	-	2
CO 4	3	2	2	2	2	-	-	-	-	-	-	2
CO 5	3	2	2	-	-	-	-	-	-	-	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test 1(Marks)	Test 2(Marks)	
Remember	10	10	20
Understand	30	30	60
Apply	10	10	20

Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Describe fundamentals of image representation

2. Discuss standards for representing colour images

3. Explain applications of image processing.

Course Outcome 2 (CO2)

1. Apply various image transforms

2. Compare lossy and lossless image compression

3. Compare techniques and standards for image compression

Course Outcome 3(CO3):

1. Compare various image transformation functions

2. Discuss the use of image transformation functions for image enhancement in spatial domain

3. Explain the basics of spatial filtering

Course Outcome 4 (CO4):

1. Differentiate various image enhancement methods in frequency domain

2. Differentiate smoothing and sharpening filters in frequency domain

3. Illustrate the concept of homomorphic filtering for image enhancement

Course Outcome 5 (CO5):

1. Discuss the concept of image segmentation

2. Explain various approaches for image segmentation

3. Discuss various morphological operations in digital image processing

Model Question paper

Course Code: ITT362

Course Name: DIGITAL IMAGE PROCESSING

Max.Marks:100
Duration: 3 Hours

PART A
Answer all Questions. Each question carries 3 Marks

1. Define m-adjacency
2. Explain RGB model for colour image representation
3. Compute the value of the underlined pixel after 5x5 median filter for the image [2 1 3 4 5 ; 1 1 4 2 3 ; 2 1 <u>0</u> 1 2 ; 5 1 4 3 1 ; 4 3 1 2 0]
4. Write mathematical equation for power law gray level transformations
5. Differentiate between ideal low pass and high pass filter in frequency domain
6. Describe the concept of homomorphic filtering for image enhancement
7. Explain Huffman coding with simple example.
8. Discuss about region growing based image segmentation.
9. Differentiate between dilation and erosion
10. Explain closing and opening morphological operations with suitable examples

Part B
Answer any one Question from each module. Each question carries 14 Marks

11. a) Explain the image formation model and briefly explain significance of sampling and quantization. 7 Marks

b) List and explain three areas in which digital image processing is widely used. 7 Marks

OR

12 a)With a neat block diagram, explain the fundamental steps in digital image processing.

. 6 Marks

b) Describe the terms i) brightness ii)contrast iii) hue iv) saturation 8 Marks

13. a) Define the 2D Walsh transform function and construct the Walsh basis matrix for N = 4 6 Marks

b) A file contains the following characters with the frequencies as shown. If Huffman Coding is used for data compression, determine-

1. Huffman Code for each character
2. Average code length
3. Length of Huffman encoded message (in bits)

characters	frequencies
a	10
e	15
i	12
o	3
u	4
s	13
t	1

8 Marks

OR

14 a) Explain JPEG image compression with help of a block schematic

7 marks

b) Compute the inverse 2D DFT of the transform coefficient given below 7 Marks

$$\begin{matrix} 16 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{matrix}$$

15. a) Explain the following methods of image enhancement in spatial domain.
i) Power Law Transformation.
ii) Gray level slicing 8 marks

b) Differentiate between linear and nonlinear spatial filter. 6 Marks

OR

16. a) Explain the various sharpening filters used in spatial domain. 7 marks

b) Explain the following grey level transformation functions
i) image negatives
ii) Log Transformation 7 Marks

17. a) Explain Butterworth filters for image smoothening and image sharpening. 7 Marks

b) Write short note on
(i) Averaging filter (ii) Weighted Averaging filter. 6 marks

OR

18. a)Describe about unsharp masking and highboost filtering in frequency domain. 7 Marks

b)What do you mean by histogram of an image? Explain about the histogram of basic image types. 8 Marks

19. a) Explain region split and merge technique for image segmentation 7 Marks

b) Write a short note on edge detection. 8 Marks

OR

20. a) Discuss the concept of boundary segments. 7 Marks

b) Define image gradient and explain its uses in edge detection. 7 Marks

Module 1: Introduction to Image processing (7 Hours)
Introduction to Image processing: Fundamental steps in image processing; Pixels; Image sampling and quantization; Representation of digital images; Basic relationship between pixels; Brightness, contrast, hue, saturation; Colour image fundamentals-RGB, CMY, HIS models; Applications of Image Processing
Module 2: Image transforms (7 Hours)
Image transforms - Discrete Fourier Transform; Discrete Cosine Transform; Walsh Transform; Hadamard Transform. Image compression –Fundamental concepts of image compression – Compression models. Lossless Compression – Huffman Coding, Run length coding. Lossy compression – Transform coding, JPEG standard
Module 3: Image Enhancement in spatial domain (7 Hours)
Image Enhancement in spatial domain: Basic Gray Level Transformation functions – Image Negatives; Log Transformations; Power-Law Transformations. Piecewise-Linear Transformation Functions: Contrast Stretching; Histogram Processing–Equalization. Basics of Spatial Filtering – Smoothing: Smoothing Linear Filters; Ordered Statistic Filters; Sharpening Filters: Laplacian
Module 4: Image Enhancement in Frequency Domain (7 Hours)
Image Enhancement in Frequency Domain: Basics of Filtering in Frequency Domain, Filters - Smoothing Frequency Domain Filters : Ideal Low Pass Filter; Butterworth Low Pass Filter; Sharpening Frequency Domain Filters: Ideal High Pass Filter; Butterworth High Pass Filter; Homomorphic Filtering.
Module 5 : Image Segmentation (7 Hours)
Image Segmentation: Pixel-Based Approach- Multi-Level Thresholding, Local Thresholding, Threshold Detection Method; Region-Based Approach- Region Growing Based Segmentation, Region Splitting, Region Merging, Split and Merge, Edge Detection - Edge Operators. Morphological Operations, Basics of Set Theory; Dilation and Erosion - Dilation, Erosion; Structuring Element; Opening and Closing;

Text Books

1. Rafael C. Gonzalez, Richard E. Woods, Digital Image Processing (English) 3rd Edition, Pearson India, 2013.

2. S Jayaraman, S Esakkirajan and T Veerakumar, Digital Image Procesing, McGraw Hill Education , 2009.

Reference Books

1.A K. Jain, Fundamentals of digital image processing, Prentice Hall of India, 1989

2.AlBovik, The Essential Guide to Image Processing, Academic Press, 2009

3.MilanSonka, Vaclav Hlavac and Roger Boyle, Image Processing, Analysis, and Machine Vision, Thomson Learning, 2008.

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction to Image processing**	**7 Hours**
1.1	Fundamental steps in image processing	1 Hour
1.2	Pixels; Image sampling and quantization	1 Hour
1.3	Representation of digital images;	1 Hour
1.4	Basic relationship between pixels; Brightness, contrast, hue, saturation	2 Hours
1.5	Colour image fundamentals-RGB, CMY, HIS models	1 Hour
1.6	Applications of Image Processing	1 Hour
2	**Image transforms**	**7 Hours**
2.1	Discrete Fourier Transform, Discrete Cosine Transform;	1 Hour
2.2	Walsh Transform; Hadamard Transform	1 Hour
2.3	Image compression –Fundamental concepts of image compression	1 Hour
2.4	Compression models. Lossless Compression , Huffman Coding, Run length coding	2 Hours
2.5	Lossy compression – Transform coding	1 Hour
2.6	JPEG standard	1 Hour
3	**Image Enhancement in spatial domain**	**7 Hours**
3.1	Basic Gray Level Transformation functions – Image Negatives;	1 Hour
3.2	Log Transformations; Power-Law Transformations	1 Hour
3.3	Piecewise-Linear Transformation Functions, Contrast Stretching;	2 Hours
3.4	Histogram Processing–Equalization	1 Hour
3.5	Basics of Spatial Filtering – Smoothing, Smoothing Linear Filters; Ordered Statistic Filters, Sharpening Filters: Laplacian	2 Hours
4	**Image Enhancement in Frequency Domain**	**7 Hours**
4.1	Basics of Filtering in Frequency Domain	1 Hour
4.2	Filters -Smoothing Frequency Domain Filters	1 Hour
4.3	Ideal Low Pass Filter; Butterworth Low Pass Filter	1 Hour
4.4	Sharpening Frequency Domain Filters: Ideal High Pass Filter	2 Hours
4.5	Butterworth High Pass Filter	1 Hour

4.6	Homomorphic Filtering.	1 Hour
5	**Image Segmentation**	**7 Hours**
5.1	Pixel-Based Approach- Multi-Level Thresholding, Local Thresholding, Threshold Detection Method	1 Hour
5.2	Region-Based Approach- Region Growing Based Segmentation, Region Splitting, Region Merging, Split and Merge	2 Hours
5.3	Edge Detection - Edge Operators. Morphological Operations, Basics of Set Theory;	2Hours
5.4	Dilation and Erosion - Dilation, Erosion; Structuring Element; Opening and Closing;	2 Hours

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT372	SEMANTIC WEB	PEC	2	1	0	3

Preamble: This course is intended to make the students capable of understanding the limitations of today's web. Keyword-based search will be replaced by semantic query answering. It covers handling RDF Schema Language and Querying RDF document using SPARQL and developing the basic idea of OWL Language.

Prerequisite: ITT301 Web application Development

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome	Bloom's Category Level
CO 1	Demonstrate the concepts of semantic web technologies and semantic web architecture	Level 2: Understand
CO 2	Explain the use of XML in Semantic Web	Level 2: Understand
CO 3	Discuss RDF and OWL Languages.	Level 2: Understand
CO 4	Interpret Logic and Inference	Level 3: Apply
CO 5	Develop applications of semantic web and make use of the concepts of ontology engineering .	Level 3: Apply

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	1	2	2	3	2	1	-	-	1	-	-	2
CO 2	2	2	1	1	2	2	-	-	1	-	-	-
CO 3	1	2	1	2	3	1	-	-	-	2	-	-
CO 4	2	2	1	3	1	2	-	2	-	-	-	-
CO 5	2	3	1	2	1	2	1	-	1	-	1	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test 1 (Marks)	Test 2(Marks)	
Remember	5	5	10
Understand	30	30	60
Apply	15	15	30
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 Hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks

Continuous Assessment Test (2 numbers) : 25 marks

Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Define Semantic Web.

2. Describe about Software Agents?

3. Explain about semantic web layers?

Course Outcome 2 (CO2):

1. What is a Statement in RDF?

2. Explain Core Classes and Core Properties in RDF.

3. State the Reification in Statements?

Course Outcome 3(CO3):

1. Describe in detail about OWL Full, OWL DL, and OWL Lite.

2. Define Thing Class and Nothing Class in OWL.

3. Explain the concept of OWL DLP with Diagram.

Course Outcome 4 (CO4):

1. State Defeasible rules and Competing Rules.

2. Explain the importance of Logic.

3. Describe about First-Order Interpretation of Goals in Monotonic Rules. Also explain Parameterized Witnesses and Ground Witnesses.

Course Outcome 5 (CO5):

1. State the Elsevier Problem? Explain the contribution of Semantic web Technology in Elsevier Problem.

2. Describe about Main stages in Ontology Development.

3. Explain about Ontology Mapping.

Model Question paper

Course Code: ITT372
Course Name: SEMANTIC WEB

Max. Marks: 100 Duration: 3 Hours

PART A

(Answer all Questions. Each question carries 3 Marks)

1.State the Problems of Keyword Based Search Engines

2. What is an XML Prolog?

3.Explain Resource, Properties in RDF with Example.

4.What is a Statement in RDF?

5.Write notes on Meta Classes.

6. Describe about OWL DL

7. Write notes on SWRL

8.Define Variables, Constants, Predicates and Function Symbols with Example.

9.Restate about Elseveir Problem.

10. List out the points of DAML-S and OWL-S.

Part B

(Answer all Questions. Each question carries 14 marks)

11 a) Explain about Semantic Layers with neat Diagram. (7 Marks)
b) Explain about Semantic Web Technologies. (7Marks)

OR

12. a) Write the Examples of Semantic Web. (7 Marks)

b) Describe about Structuring of XML Document this using XML Schema (7 Marks)

13.a) Describe in detail about OWL Full, OWL DL, and OWL Lite. (6 Marks)

b) Explain about Querying in SPARQL. (8 Marks)

OR

14. a) Summarize the Constraints of axiomatic semantics for RDF and RDF Schema (7 Marks)

b) Write the RDF Schema of "Lecturer is a subclass of Academic Staff Member. (7 Marks)

15. a) Explain three Sublanguages of OWL (7 Marks)

b) Describe about the Requirements of Ontology Languages? (7 Marks)

OR

16. a) State Defeasible rules and Competing Rules (6 Marks)

b) Explain the two kinds of Properties in OWL with Example. (8 Marks)

17 a) Explain about the Semantics of Monotonic Rules (7 Marks)

b) Explain about Description Logic Programs. (7 Marks)

OR

18 a) How would you apply Priorities are used to resolve some conflicts between
19 Non-Monotonic Rules? (7 Marks)

b) Describe about First-Order Interpretation of Goals in Monotonic Rules. Also explain Parameterized Witnesses and Ground Witnesses. (7 Marks)

20 a) What is Elsevier Problem? How would you solve this Problem by using Semantic Web Technology? (6 Marks)

b) Discuss about main stages of ontology development. (8 Marks)

OR

21 a) Explain the contribution of Semantic Web Technology in E-Learning.(7 Marks)

b) Restate about the four Sub ontologies of OWL-S Ontology. (7 Marks)

Syllabus

Module 1: Introduction and Semantic Web Technologies (8 Hours)
The Semantic Web Vision, Today's Web, From Today's Web to the SemanticWeb: Examples, Semantic Web Technologies, A Layered Approach, Structured Web Document in XML, The XML Language, Structuring, Namespace.
Module 2: RDF Schema Language (6 Hours)
Describing Web Documents in RDF, RDF: XML-Based Syntax,RDF Schema, An Axiomatic Semantics for RDF and RDF Schema, Querying in RQL
Module 3: Web Ontology Language(OWL)(6 Hours)
Web Ontology Language(OWL), Examples, OWL in OWL, Future Extensions
Module 4: Logic and Inference (6 Hours)
Logic and Inference:Rules , Example of Monotonic Rules: Family Relationships , Monotonic Rules Syntax and Semantics, Nonmonotonic Rules: Motivation, Syntax and Example, Rule .Markup in XML
Module 5: Applications and Ontology Engineering(9 Hours)
Applications: Horizontal Information Products at Elsevier, e-Learning, WebServices **Ontology Engineering:** Constructing Ontologies Manually, Reusing ExistingOntologies, Ontology Mapping, On-To- Knowledge Semantic Web Architecture

Text Books

1. Grigoris Antoniou, Frank Van Harmelon, "A Semantic Web Primer", Second Edition , The MITPress.
2. J. Davies, D. Fensel, and F. van Harmelen. Towards the Semantic Web:Ontology-Driven Knowledge Management, New York, Wiley, 2003.

Reference Books

1.Natalya. F. Noy and Deborah L. McGuinness, Ontology Development 101: AGuide to Creating Your First Ontology, http://protege.stanford.edu/publications/ontology_development/ontology101.pd

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction and Semantic Web Technologies**	**8 Hours**
1.1	From Today's Web to the Semantic Web	2 hours
1.2	The Semantic Web Impact	3 hours
1.3	Semantic Web Technologies	3 hours
2	**RDF Schema Language**	**6 Hours**
2.1	XML Basics	2 hours
2.2	RDF Schema Language	2 hours
2.3	An Axiomatic Semantics for RDF and RDF Schema	2 hours
3	**Web Ontology Language(OWL)**	**6 Hours**
3.1	Examples of OWL	1 hours
3.2	OWL Sublanguages	3 hours
3.3	OWL in OWL	2 hours
4	**Logic and Inference**	**6 Hours**
4.1	Monotonic Rules	2 hours
4.2	Non Monotonic Rules	3 hours
4.3	Rule Markup in XML	1 hours
5	**Applications and Ontology Engineering**	**9 Hours**
5.1	Elsevier, e-Learning, Web Services	4 hours
5.2	Constructing Ontologies Manually	3 hours
5.3	Reusing Existing Ontologies	2 hours

SEMESTER VI

MINOR

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT382	ANDROID PROGRAMMING	VAC	3	1	0	4

Preamble: The syllabus is prepared with the intended to deliver students the elementary concepts of Android App Development and equip them to code android application built over those concepts. It also introduces to them advanced level areas like event driven programming with Android.

Prerequisite: Java Programming (ITT281), Database Management (ITT282).

Course Outcome (CO): After completion of the course, the student will be able to

CO No.	Course Outcome	Bloom's Category Level
CO1	Summarize about different technologies used in mobile.	Level2: Understand
CO2	Summarize Kotlin Programming Language Concepts	Level 2: Understand
CO3	Summarize about Android Framework	Level 2: Understand
CO4	Build basic android applications.	Level 3: Apply
CO5	Build android apps using advanced android concepts.	Level 3: Apply

Mapping of Course Outcomes with Program Outcomes

POs / COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
CO1	1	2	2	-	1	-	-	-	-	-	-	-
CO2	1	1	2	1	-	-	-	-	-	-	-	-
CO3	3	1	1	-	-	-	-	-	-	-	-	-
CO4	3	1	1	1	-	-	-	-	-	-	-	-
CO5	3	1	1	2	2	-	-	-	-	-	-	-

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests (Marks)		End Semester Examination (Marks)
	Test 1 (Marks)	Test 2 (Marks)	
Remember	15	15	30
Understand	25	25	50
Apply	10	10	20
Analyze			
Evaluate			
Create			

Mark Distribution

Total Marks	CIE	ESE	ESE duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain processors used in mobile.
2. Describe different sensors used in mobile.
3. Familiarization with mobile peripherals and display technologies...
4. Familiarization with mobile operating systems.
5. Explain different mobile platform architectures.

Course Outcome 2 (CO2):

1. Describe the fundamentals of Kotlin Language.
2. Explain about the conditional and looping statements.

3. Illustrate the concept of Higher Order Functions with an Example.
4. Demonstrate the concept of Classes with an example.
5. Explain ranges and collections in Kotlin programming language.

Course Outcome 3 (CO3):

1. Describe the android framework.
2. Describe AndroidManifest.xml.
3. Illustrate the concept of Activity with an example.
4. Demonstrate the concept of Services with an example.
5. Explain about Broadcast Receiver.

Course Outcome 4 (CO4):

1. Explain Android user interface.
2. Describe UI Layouts in Android.
3. Explain UI Controls in Android.
4. Illustrate the concept of Event handling.
5. Develop an Android app with appropriate UI, UI Layouts, UI Controls and Event handling.

Course Outcome 5 (CO5):

1. Explain device location access.
2. Demonstrate usage of SQLite database with an example.
3. Illustrate how data can be accessed from the internet.
4. Develop an efficient android app using Recycler View and database connectivity.

Model Question Paper

Course Code: ITT382

Course Name: ANDROID PROGRAMMING

Max.Marks:100 **Duration: 3 Hours**

PART A

(Answer all questions, each carries 3 marks)

1. What are the advantages of using ARM Processor?
2. What is AMOLED?
3. With an example explain the use of lambdas.
4. What is the concept of null safety?
5. Name any three Android Studio IDE project components and their use?
6. How notifications can be provided?
7. Describe UI Layout attributes.
8 Explain intents in Android.
9. How can an android application be published?
10. Compare and contrast different types of location access?

PART B

(Answer all questions. Each full question carries 14 marks)(5*14=70)

11 a. With a neat diagram, explain the architecture of the Android operating system. (8)

b. Mention about different sensors used in mobile devices. (6)

OR

12 a. With a neat diagram, explain the architecture of the iOS. (8)

b. What are the features of a mobile operating system? Give examples of mobile OS. (6)

13 a. Write a menu driven program to perform arithmetic operations. (10)

a. With an example explain the use of higher order functions. (4)

OR

14 a. Define class. How they can be implemented in Kotlin? (8)

b. What are the different types of inheritance in Kotlin. Give examples. (6)

15 a. Define Android activity. Explain the callbacks defined in activity class (7)

b. Explain the use and significance of AndroidManifest.xml (7)

OR

16 a. With an example illustrate how Broadcast Receiver can be created and used. (7)

b. With a neat diagram explain about Android Service lifecycle (7)

17 a. Explain about different Android Layouts. Give examples (9)

b. Illustrate how an activity can be started from another activity. (5)

OR

18 a. Explain about any four Android UI Controls. (8)

b. With an example demonstrate event handling in Android. (6)

19 a. Explain how content providers supply data from one application to another. Write the steps for creating your own content provider. Also, list out the methods needed to be overridden for the proper working of user defined content provider. (9)

b. Write an android program to access data from internet to implement a social media app with posting capability. (6)

OR

20 a. Write an android program with proper configuration files to show lists using Recycler View. (14)

Syllabus

MODULE 1: GENERAL MOBILE TECHNOLOGY AWARENESS (8 Hours)
Device hardware - Processors - ARM processors. Sensors - Base sensors, Composite Sensors, Uncaliberated, and Interaction composite sensors. Peripherals - audio and custom accessories. Display technologies - Touch, LCD, LED, OLED, AMOLED, display densities. Operating systems - Android, iOS, KaiOS, Tizen, and other operating systems. Platform Architecture: iOS and Android. App Stores – Apple App Store - Google Play. Development languages: Swift for iOS, Kotlin for Android.
MODULE 2: INTRODUCTION TO KOTLIN PROGRAMMING LANGUAGE (8 Hours)
Introduction to Kotlin - basic data types, variables, control flow - conditional statements, Null safety, loops - for loop, while loop, when expression, ranges, collections, functions, higher order functions, lambdas, inline functions. Classes, inheritance, extensions, data classes, objects, enums.
MODULE 3: INTRODUCTION TO ANDROID FRAMEWORK (11 Hours)
Introduction to android framework. Process and application Lifecycle. Introduction to Android Studio IDE. Project components - source files, resources, images, layout XML, and raw assets. Build system - Gradle, build.gradle, adding dependencies. Android app components - Activities, Services, Notifications, Broadcast Receivers, Content Providers.
MODULE 4: BUILDING BASIC ANDROID APPS (9 Hours)
Developing Basic apps using Kotlin, Activities, lifecycle, starting another Activity, Intents. Basic UI components - Layouts, TextView, EditText, Button, CheckBox, ImageView, ToggleButton, ProgressBar. Events and Event Handling.
MODULE 5: ADVANCED ANDROID CONCEPTS (9 Hours)
Showing lists using RecyclerView. Accessing device location, SQLite database, Accessing data from the Internet.

References

1. Android Studio 4.0 Development Essentials - Kotlin Edition by Neil Smyth, Payload Media, June 2020.
2. Android Programming Unleashed, B. M. Harwani, Sams, December 2012
3. ARM Architecture: https://en.wikipedia.org/wiki/ARM_architecture
4. Android Sensors: https://source.android.com/devices/sensors/sensor-types
5. Mobile OS list: https://en.wikipedia.org/wiki/Mobile_operating_system
6. Swift: https://docs.swift.org/swift-book/GuidedTour/GuidedTour.html
7. Kotlin: https://kotlinlang.org/docs/reference/basic-syntax.html
8. Android Platform Architecture: https://developer.android.com/guide/platform
9. Kotlin basics: https://kotlinlang.org/docs/reference/basic-syntax.html
10. Process & App Lifecycle: https://developer.android.com/guide/components/activities/process-lifecycle
11. Android Studio setup: https://developer.android.com/studio/intro

12. Android Studio project files: https://developer.android.com/studio/projects
13. Android App Components: https://developer.android.com/guide/components/fundamentals
14. Activities: https://developer.android.com/guide/components/activities/intro-activities
15. UI Components: https://developer.android.com/guide/topics/ui
16. Google Codelabs for app development: https://codelabs.developers.google.com/android-kotlin-fundamentals/
17. RecyclerView: https://developer.android.com/reference/kotlin/androidx/recyclerview/widget/RecyclerView
18. SQLite: https://developer.android.com/training/data-storage/sqlite
19. Location tracking: https://developer.android.com/training/location

Course Contents and Lecture Schedule

No.	Topic	No. of Lectures
1	**General Mobile Technology Awareness**	**8 Hours**
1.1	**Device hardware** - Processors - ARM processors. Sensors- Base sensors, Composite Sensors, Uncaliberated, and Interaction composite sensors. Peripherals- audio and custom accessories.	2 Hours
1.2	**Display technologies** - Touch, LCD, LED, OLED, AMOLED, display densities.	1 Hour
1.3	**Operating systems** - Android, iOS, KaiOS, Tizen, and other operating systems.	2 Hours
1.4	**Platform Architecture**: iOS and Android. App Stores – Apple App Store - Google Play. Development languages: Swift for iOS, Kotlin for Android.	3 Hours
2	**Introduction to Kotlin Programming Language**	**8 Hours**
2.1	**Introduction to Kotlin** - basic data types, variables, control flow - conditional statements, Null safety, loops - for loop, while loop, when expression, ranges, collections.	2 Hours
2.2	**Functions**- function, higher order functions, lambdas, inline functions.	3 Hours
2.3	**Classes** - class, inheritance, extensions, data classes, objects. Enums.	3 Hours
3	**Introduction to Android Framework**	**11 Hours**
3.1	**Process and application Lifecycle**.	2 Hours
3.2	**Introduction to Android Studio IDE**. Project components - source files, resources, images, layout XML, and raw assets. Build system - Gradle, build.gradle, adding dependencies. custom intents	4 Hours
3.3	**Android app components** - Activities, Services, Notifications - Toast, Broadcast Receivers, Content Providers.	5 Hours

4	**Building Basic Android apps**	**9 Hours**
4.1	**Developing Basic apps using Kotlin** - Activities, lifecycle, starting another Activity, Intents.	3 Hours
4.2	**UI components** - Layouts, TextView, EditText, Button, CheckBox, ImageView, ToggleButton, ProgressBar.	3 Hours
4.3	**Event handling** – Event, Event Listener, Event Handler.	3 Hours
5	**Advanced Android Concepts**	**9 Hours**
5.1	**RecyclerView**– concept, implementation and advantages	3 Hours
5.2	**Accessing Device Location**– Types of location access (Foreground location and Background location**)**	2 Hours
5.3	**SQLite database**– Insert, update, delete and access data	2 Hours
5.4	**Accessing data from internet**– Different methods to access data from internet.	2 Hours

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT384	INTERNETWORKING	VAC	3	1	0	4

Preamble: This subject is about the TCP/IP protocol suite and how it is used on the internet. It begins with a review of the underlying communications technologies needed for the internet. The course provides a detailed examination of IP routing, UDP, TCP, network virtualization, and label switching. Finally, internet applications andSoftware defined networking are discussed.

Prerequisite: ITT 283 Data Communication, ITT 284 Computer Networks

Course Outcomes: After the completion of the course the student will be able toA

CO No.	Course Outcome(CO)	Bloom's Category Level
CO 1	Describe internetworking concepts.	Level 2: Understand
CO 2	Identify the functions of IPv4, IPv6protocols.	Level 3: Apply
CO 3	Familiarize with internet routing architecture and routing protocol.	Level 2: Understand
CO 4	Discuss the design issues and protocols in transport layer.	Level 3: Apply
CO 5	Familiarize with label switching, application layer protocols, network virtualization and software defined networking.	Level 2: Understand

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	-	-	-	-	-	-	-	-	-	2
CO 2	3	2	1	-	-	-	-	-	-	-	-	2
CO 3	3	2	-	-	2	-	-	-	-	-	-	2
CO 4	3	2	1	-	-	-	-	-	-	-	-	2
CO 5	3	2	-	-	3	-	-	-	-	-	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test 1(Marks)	Test 2(Marks)	
Remember	10	10	20
Understand	30	30	60
Apply	10	10	20
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain the need for multiple protocols for internetworking.
2. Explain different layers in TCP/IP reference model.

Course Outcome 2 (CO2)

1. Explain IP datagram fragmentation and reassembly. Also explain different header fields affected in these cases.
2. Compare the headers of IPv4 and IPv6.

Course Outcome 3(CO3):

1. Explain characteristics and message formats in BGP.
2. Explain RIP in detail. What is slow convergence problem and how it is solved?

Course Outcome 4 (CO4):

1. Draw and explain TCP finite state machine.
2. Explain the adaptive retransmission algorithm used by TCP.
3. What is the purpose of using a pseudo-header for UDP checksum computation?
4. Explain the format of IPv6 UDP pseudo-header

Course Outcome 5 (CO5):

1. What is label switching?
2. Explain the difference between persistent and non persistent HTTP.
3. Explain DNS.

Model Question Paper

Course Code: ITT384

Course Name: INTERNETWORKING

Max. Marks: 100 Duration: 3 Hours

PART A

Answer all questions. Each question carries 3 marks. (10 * 3 = 30 Marks)

1. What is internetworking?
2. Explain the role of routers in Networks.
3. Explain the header fields in IP that are used for datagram fragmentation and reassembly
4. What are the features of connectionless delivery service?
5. Assuming that all routers and hosts are working properly and that all software in both is free of all errors, is there any chance, however small, that a packet will be delivered to the wrong destination?
6. Explain the purpose of exterior gateway routing protocols
7. How does VPN protect user data from external access?
8. A TCP machine is sending full windows of 65535 bytes over a 1 Gbps channel that has a 10 msec one way delay. What is the maximum throughput achievable?
9. What is QoS?
10. What is SDN?

Part B

Answer all questions. Each question carries 14 marks. (5 * 14 = 70 Marks)

11 List and explain the main features of all the seven layers of the ISO/OSI reference model and compare it with TCP/IP Model.

OR

12 Explain in detail about IPv6 addressing

13 Explain the format of IPv4 and IPv6 datagram.

OR

14 Explain in detail about fragmentation and reassembly of IPv4 datagram

15 What is BGP? Explain the characteristics and message formats of BGP.

OR

16 What is slow convergence problem? How can it be solved?

17 Explain in detail about TCP segment format.

OR

18 What is congestion? Explain in detail about TCP congestion control.

19 Explain triangular routing problem

OR

20 Explain in detail about RTP and RSVP

Syllabus

Module 1: Introduction to Internetworking(8 Hours)
Introduction & Overview – Motivation for internetworking, TCP/IP internet, Internet Services. Internetworking Concept And Architectural Model – Introduction, Application-Level Interconnection, Network-Level Interconnection, Properties Of The Internet, Internet Architecture, Interconnection Of Multiple Networks With IP Routers. **Protocol Layering**- Introduction, The Need For Multiple Protocols, The Conceptual Layers Of Protocol Software, Functionality Of The Layers, ISO 7-Layer Reference Model, The TCP/IP 5-Layer Reference Model, IPv6 Addressing
Module 2: Network Layer (8Hours)
Internet Protocol: Connectionless Datagram Delivery (IPv4, IPv6) – Introduction, Connectionless Delivery System Characteristics, Purpose and Importance Of The Internet Protocol, The IP Datagram, Datagram Type Of Service And Differentiated Services, Datagram Encapsulation, Datagram Size, Network MTU and Fragmentation, Datagram Reassembly, Header Fields Used For Datagram Reassembly, Time To Live (IPv4) And Hop Limit (IPv6), Optional IP Items, Options Processing During Fragmentation.
Module 3: Routing (10Hours)
Routing Architecture – Cores, Peers and Algorithms, Routing among Autonomous system – BGP - The Scope Of A Routing Update Protocol, Determining A Practical Limit On Group Size, Autonomous System Concept, Exterior Gateway Protocols And Reachability, BGP Characteristics, BGP Functionality And Message Types, Routing Within An Autonomous System - Introduction, Static Vs. Dynamic Interior Routes, Routing Information Protocol (RIP), Slow Convergence Problem, Solving The Slow Convergence Problem, The Disadvantage Of Using Hop Counts, Delay Metric (HELLO), Delay Metrics, Oscillation, And Route Flapping, The Open SPF Protocol (OSPF).
Module 4: Transport Layer (9Hours)
User Datagram protocol – UDP message format, Interpretation of UDP checksum, UDP checksum computation and pseudo header, IPv4 and IPv6 UDP pseudo header format, UDP encapsulation and protocol layering, reserved and available port numbers. **Reliable stream transport service** – need for reliable service, properties of reliable service, reliability, sliding window paradigm, transmission control protocol, Segments, streams and sequence numbers, variable window size and flow control, TCP segment format, TCP options, TCP checksum computation, acknowledgements, retransmissions and timeouts, Karn's algorithm and timer backoff, response to congestion, random early detection, establishing and closing a TCP connection, TCP finite state machine
Module 5: Application Layer (10 Hours)
Label switching, flows and MPLS. Network Virtualization: VPNs, NATs, And Overlays. Mobility and Mobile IP – Introduction, Mobility, Addressing, Routing, Mobility via host address change and changes in datagram forwarding, Mobile IP and operation, Mobile IPv4 addressing, IPv4 foreign agent discovery and registration, communication within an IPv4

foreign agent, IPv6 mobility support,datagram transmission, reception and tunnelling. Voice and video over IP (RTP, RSVP, QoS), Software defined networking – Routes, paths, connections, traffic engineering and control of path selection, connection oriented networks, routing overlays, SDN, separation of data and control plane, SDN architecture and external controllers, SDN across multiple devices, implementing SDN with conventional switches, Openflow technology.

Text Books

1.Douglas E Comer, "Internetworking with TCP/IP Principles, Protocol, and Architecture" , Volume I, 6th Edition, Pearson Education, 2013
2.Andrew S. Tanenbaum, "Computer Networks", Prentice Hall, 5th Edition
3.James F Kurose, Keith W Ross, Computer Networking: A top Down Approach featuring the Internet, Pearson Education, 3rd Edition

Reference Books

1. Behrouz A Forouzan, TCP/IP Protocol Suite, Fourth Edition

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction to Internetworking-**	**8 Hours**
1.1	Introduction & Overview – Motivation for internetworking, TCP/IP internet, Internet Services. Internetworking Concept And Architectural Model – Introduction, Application-Level Interconnection, Network-Level Interconnection, Properties Of The Internet, Internet Architecture, Interconnection Of Multiple Networks With IP Routers.	2Hours
1.2	Protocol Layering- Introduction, The Need For Multiple Protocols, The Conceptual Layers Of Protocol Software, Functionality Of The Layers, ISO 7-Layer Reference Model.	2Hours
1.3	The TCP/IP 5-Layer Reference Model	2Hours
1.4	IPv6 Addressing	2 Hours
2	**Network Layer-**	**8 Hours**
2.1	Internet Protocol: Connectionless Datagram Delivery (IPv4, IPv6) – Introduction, Connectionless Delivery System Characteristics, Purpose And Importance Of The Internet Protocol, The IP Datagram, Datagram Type Of Service And Differentiated Services, Datagram Encapsulation, Datagram Size	5Hours
2.2	Network MTU and Fragmentation, Datagram Reassembly, Header Fields Used for Datagram Reassembly, Time To Live (IPv4) and Hop Limit (IPv6),Optional IP Items, Options Processing During Fragmentation.	3Hours
3	**Routing -**	**10 Hours**
3.1	Routing Architecture – Cores, Peers and Algorithms, Routing	2 Hours

	among Autonomous system – BGP - The Scope Of A Routing Update Protocol, Determining A Practical Limit On Group Size, Autonomous System Concept	
3.2	Exterior Gateway Protocols And Reachability, BGP Characteristics, BGP Functionality And Message Types	2 Hours
3.3	Routing Within An Autonomous System - Introduction, Static Vs. Dynamic Interior Routes	1Hour
3.4	Routing Information Protocol (RIP), Slow Convergence Problem, Solving The Slow Convergence Problem, The Disadvantage Of Using Hop Counts	2 Hours
3.5	Delay Metric (HELLO), Delay Metrics, Oscillation, And Route Flapping	1 Hour
3.6	The Open SPF Protocol (OSPF).	2 Hours
4	**Transport Layer-**	**9 Hours**
4.1	**User Datagram protocol –** UDP message format, Interpretation of UDP checksum, UDP checksum computation and pseudo header, IPv4 and IPV6 UDP pseudo header format, UDP encapsulation and protocol layering, reserved and available port numbers.	3 Hours
4.2	**Reliable stream transport service** – need for reliable service, properties of reliable service, reliability, sliding window paradigm, transmission control protocol, Segments, streams and sequence numbers, variable window size and flow control	2 Hours
4.3	TCP segment format, TCP options, TCP checksum computation, acknowledgements, retransmissions and timeouts	1 Hours
4.4	Karn's algorithm and timer backoff, response to congestion, random early detection	1 Hour
4.5	Establishing and closing a TCP connection, TCP finite state machine	2 Hours
5	**Application Layer-**	**10 Hours**
5.1	Label switching, flows and MPLS. Network Virtualization: VPNs, NATs, And Overlays	2 Hours
5.2	Mobility and Mobile IP – Introduction, Mobility, Addressing, Routing, Mobility via host address change and changes in datagram forwarding, Mobile IP and operation, Mobile IPv4 addressing, IPv4 foreign agent discovery and registration, communication within an IPv4 foreign agent, IPv6 mobility support, datagram transmission, reception and tunnelling.	2Hours
5.3	Voice and video over IP (RTP, RSVP, QoS	2Hours
5.4	Software defined networking – Routes, paths, connections, traffic engineering and control of path selection, connection oriented networks, routing overlays, SDN, separation of data and control plane	1Hour
5.5	SDN architecture and external controllers, SDN across multiple devices, implementing SDN with conventional switches.	1 Hour
5.6	Openflow technology	2 Hours

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT386	PRINCIPLES OF SOFTWARE QUALITY ASSURANCE	VAC	3	1	0	4

Preamble: The syllabus is prepared with the view of preparing the Engineering Graduates capable ofunderstanding essential principles of software quality assurance.

Prerequisite: Basicsof Software Engineering

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO1	Summarize role of software quality assurance	Level 2: Understand
CO2	Identify the factors affecting software quality	Level 2: Understand
CO3	Explain the steps to assure software quality	Level 2: Understand
CO4	Explain Software quality metrics	Level 2: Understand
CO5	Discuss the role of management in quality assurance	Level 2: Understand
CO6	Illustrate CASE tools and their effect on software quality	Level 2: Understand

Mapping of course outcomes with program outcomes

POs / COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	2	1	-	3	2	-	2	3	1	2	1
CO2	2	3	3	1	3	-	-	1	2	2	3	1
CO3	2	1	1	-	3	1	-	1	-	1	-	1
CO4	-	1	-	1	-	1	-	1	3	3	2	1
CO5	1	3	-	1	2	-	-	-	-	-	2	1
CO6	1	-	2	1	-	3	-	3	-	-	2	1

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination Marks
	Test 1(Marks)	Test 2(Marks)	
Remember	10	10	20
Understand	40	40	80
Apply			
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part Acontain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. List out the importance of software quality assurance.
2. Explain change control.
3. Describe the elements of Software Configuration Management.
4. Illustrate the steps to develop and implement a Software Quality Assurance Plan.

Course Outcome 2 (CO2):

1. Classify software requirements into software quality factors.
2. Differentiate Verification, validation and qualification.
3. Discuss the components of the software quality assurance system
4. Illustrate the Integration of quality activities in the project life cycle.

Course Outcome 3 (CO3):

1. Compare and contrast team review methods.
2. Illustrate the types of external participants.
3. Explain software testing strategies and implementation.
4. Discuss the process of Assuring the quality of software maintenance components

Course Outcome 4 (CO4):

1. Analyze the basis for economic analysis of architecture.
2. Identify the objectives of quality measurement.
3. Describe Team Software Process (TSP) and Personal Software Process (PSP)
4. Explain the process of Measuring Process Improvement.

Course Outcome 5 (CO5):

1. Explain the classic model of cost of software quality.
2. Illustrate the models of Cost of software quality

Course Outcome 6 (CO6):

1. Analyse the contribution of CASE tools to improved project Management.
2. Explain the effect of CASE tools on software quality.

Model Question Paper

Course Code: ITT386

Course Name: PRINCIPLES OF SOFTWARE QUALITY ASSURANCE

Max.Marks:100 **Duration: 3 Hour**

PART A

(Each Question carries 3 Marks) (10*3=30)

1. List out the importance of software quality assurance
2. Explain change control.
3. Classify software requirements into software quality factors.
4. Differentiate Verification, validation and qualification.
5. Compare and contrast team review methods.
6. Illustrate the types of external participants.
7. Analyze the basis for economic analysis of architecture.
8. Identify the objectives of quality measurement.
9. Explain the classic model of cost of software quality.
10. Analyse the contribution of CASE tools to improved project Management.

PART B

(Answer all Questions. Each question carries 14 marks) (5*14=70)

11. Describe the elements of Software Configuration Management.

OR

12. Illustrate the steps to develop and implement a Software Quality Assurance Plan.
13. Discuss the components of the software quality assurance system

OR

14. Illustrate the Integration of quality activities in the project life cycle.
15. Explain software testing strategies and implementation.

OR

16. Discuss the process of Assuring the quality of software maintenance components

17. Describe Team Software Process (TSP) and Personal Software Process (PSP)

OR

18. Explain the process of Measuring Process Improvement.
19. Illustrate the models of Cost of software quality

OR

20. Explain the effect of CASE tools on software quality.

Syllabus

Module 1: Introduction to software quality (9 Hours)
Quality Assurance Framework, Elements of Software Configuration Management, Quality Standards
Module 2: Software quality factors (9 Hours)
Software quality factors, The components of the software quality assurance system, Integrating quality activities in the project life cycle
Module 3: Assuring the quality (9 Hours)
Reviews, Software testing – strategies, implementation, Assuring the quality of software maintenance components, Assuring the quality of external participants' contributions.
Module 4: Software quality metrics (9 Hours)
Objectives of quality measurement, Software Quality Indicators, Software Quality Fault Prediction
Module 5: Management and its role in software quality assurance (9 Hours)
Costs of software quality, Management and its role, CASE tools and their effect on software quality

Text Books

1. Handbook of Software Quality Assurance, Fourth Edition, G. Gordon Schulmeyer, ARTECH HOUSE, INC., 2008. ISBN-13: 978-1-59693-186-2.
2. Software Testing and Continuous Quality Improvement, William E. Lewis, 3rd Ed. CRC Press, 2016, ISBN 9781420080735

Reference Books

1. Software Testing: Principles, Techniques and Tools, M. G. Limaye, TMH, 2017
2. Foundations of Software Testing, Dorothy Graham, Erik van Veenendaal, Isabel Evans, Rex Black, Cengage Learning, 3rd
3. Software Testing: A Craftsman's Approach, Paul C. Jorgenson, CRC Press, 4th, 2017

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Introduction to software quality**	**9 Hours**
1.1	Quality Assurance Framework:- Quality, Prevention versus Detection , Verification versus Validation	1 Hour
1.2	Software Quality Assurance, Components of Quality Assurance, Software Testing, Quality Control, Software Configuration Management	2 Hours
1.3	Elements of Software Configuration Management, Component Identification, Version Control, Configuration Building, Change Control	2 Hours
1.4	Software Quality Assurance Plan, Steps to Develop and Implement a Software Quality Assurance Plan	2 Hours
1.5	Quality Standards	2 Hours
2	**Software quality factors**	**9 Hours**
2.1	**Software quality factors** The need for comprehensive software quality requirements, Classifications of software requirements into software quality factors, Product operation software quality factors , Product revision software quality factors, Product transition software quality factors, Alternative models of software quality factors, Software compliance with quality factors	3 Hours
2.2	**The components of the software quality assurance system – overview** The SQA system – an SQA architecture, Pre-project components, Software project life cycle components, Infrastructure components for error prevention and improvement, Management SQA components, SQA standards, system certification, and assessment components, Organizing for SQA – the human components, Considerations guiding construction of an organization's SQA system	3 Hours
2.3	**Integrating quality activities in the project life cycle** Classic and other software development methodologies, Factors affecting	3 Hours

	intensity of quality assurance activities in the development process, Verification, validation and qualification, A model for SQA defect removal effectiveness and cost	
3	**Assuring the quality**	**9 Hours**
3.1	**Reviews** Review objectives, Formal design reviews (DRs) , Peer reviews , A comparison of the team review methods, Expert opinions	1 Hour
3.2	**Software testing – strategies** Definition and objectives, Software testing strategies , Software test classifications , White box testing , Black box testing	2 Hours
3.3	**Software testing – implementation** The testing process, Test case design, Automated testing, Alpha and beta site testing programs	2 Hours
3.4	**Assuring the quality of software maintenance components** The foundations of high quality, Pre-maintenance software quality components, Maintenance software quality assurance tools	2 Hours
3.5	**Assuring the quality of external participants' contributions** Types of external participants , Risks and benefits of introducing external participants , Assuring quality of external participants' contributions: objectives , SQA tools for assuring the quality of external participants' contributions	2 Hours
4	**Software quality metrics**	**9 Hours**
4.1	Objectives of quality measurement, Classification of software quality metrics, Process metrics Product metrics, Implementation of software quality metrics , Limitations of software metrics	2 Hours
4.2	Software Quality Indicators, Practical Software and Systems Measurement (PSM) , CMMI Measurement and Analysis , CMMI Higher Maturity Measurements	2 Hours
4.3	Practical Implementations, Hewlett Packard, Quantitative SQA, Pragmatic Quality Metrics, Effectiveness Measure , Team Software Process (TSP) , and Personal Software Process (PSP).	3 Hours
4.4	Software Quality Fault Prediction, Measuring Process Improvement Using Stoplight Charts , Six Sigma , Project Managers Control Panel , Predicting Software Quality	2 Hours
5	**Management and its role in software quality assurance**	**9 Hours**
5.1	**Costs of software quality** Objectives of cost of software quality metrics , The classic model of cost of software quality , An extended model for cost of software quality , Application of a cost of software quality system , Problems in the application of cost of software quality metrics	3 Hours

5.2	**Management and its role in software quality assurance** Top management's quality assurance activities , Department management responsibilities for quality assurance , Project management responsibilities for quality assurance	3 Hours
5.3	**CASE tools and their effect on software quality** CASE tool, The contribution of CASE tools to software product quality, The contribution of CASE tools to software maintenance quality,The contribution of CASE tools to improved project Management.	3 Hours

SEMESTER VI

HONOURS

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT394	DESIGN AND ANALYSIS OF NETWORKS	VAC	3	1	0	4

Preamble: This course covers problems in design and analysis of computer networks.

Prerequisite: ITT272 Mathematical Foundation for Networking, ITT393 Wireless Communication

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO 1	Identify the requirements of a network	Level 2: Understand
CO 2	Identify the characteristics of traffic flow on a network and design network architecture.	Level 3: Apply
CO 3	Describe the fundamental concepts about addressing and routing for IP networks and implement network management and the network managementarchitecture.	Level 3: Apply
CO 4	Describethe mechanisms to achieve performance and understand the security mechanisms such as physicalsecurity, protocol and application security, encryption/decryption, and perimeterand remote access security	Level 2: Understand
CO 5	Discuss the need to evaluate the vendor or service provider and come up with traceable blueprints to suit the need of the vendor or service provider	Level 2: Understand

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	3	3	2	-	-	-	-	-	-	-	3
CO 2	3	3	3	2	-	-	-	-	-	-	-	2
CO 3	3	3	3	2	-	-	-	-	-	-	-	2
CO 4	3	3	3	2	-	-	-	-	-	-	-	2
CO 5	3	3	3	2	-	-	-	-	-	-	-	3

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	Test 1(Marks)	Test 2(Marks)	
Remember	10	10	10
Understand	20	20	50
Apply	20	20	40
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance	: 10 marks
Continuous Assessment Test (2 numbers)	: 25 marks
Assignment/Quiz/Course project	: 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Why is requirements analysis important to network architecture and design? Give three reasons.

2. Your customer is a hospital that wants to upgrade its LAN. Develop a questionnaireto gather requirements from users, hospital management, and staff. What kinds of questions would you ask to better understand their environment?

Course Outcome 2 (CO2)

1. Develop a flow model for real-time/near-real-time flows. How would you characterize the flows for this model? What are likely data sources and sinks? Apply your model to an online transaction processing (OLTP) application.

2. What are the differences between the LAN/MAN/WAN and Access/Distribution/Core architectural models? Under what conditions might each be applied to anetwork?

Course Outcome 3(CO3):

1. Subnet 136.178.0.0 into 16 subnets. Show the binary and dotted-decimal forms ofeach subnet, as well as the subnet mask.
2. Which of the following are end-to-end characteristics? Per-link/per-network/per-elementcharacteristics?
   a. Round-trip delay between a network management device and a computing server
   b. Percent utilization of the buffers in an IP router
   c. Maximum throughput between two devices (client and server) using a client-server application
   d. Bit error rate (BER) of a DS3 link

Course Outcome 4 (CO4):

1. List four types of problems that the performance architecture addresses. Giveexamples of each type of problem
2. Explain threat analysis. Give an example of a threat analysis worksheet for an organization

Course Outcome 5 (CO5):

1. Compare differentiated service and integrated service.
2. What are the mechanisms used for traffic management? Explain.

Model Question Paper

Course Code:ITT394
Course Name: DESIGN AND ANALYSIS OF NETWORKS
Max.Marks:100 **Duration: 3 Hour**

PART A

Answer all questions. Each question carries 3 marks. (10 * 3 = 30 Marks)

1. Give an example of a mission-critical application for each of these three environments: government, military, commercial. Why would each application be considered mission-critical?
2. Describe two ways to make an uptime requirement of 99.999% more precise.
3. Which flow models apply to each set of flows described below?

a. Users on the Internet accessing the same Web server

b. Forty workstations processing batch jobs overnight, managed by a central mainframe

c. Email use across the Internet

4. Give examples of physical and functional entities that you would use as building blocks for a network.

5. What is the class of each address? Give the default subnet mask.

    a. 192.12.102.0

    b. 10.20.30.100

    c. 130.5.77.15

6. Give any two ways that the management data collected via SNMP counters from each of the routers could be verified.

7. Give two examples each of the requirements which indicate single-tier performance and which indicate Multi-tier performance?

8. What are the common problems that are addressed by the performance architecture?

9. What are ad hoc design decisions? Give an example of an ad hoc design decision.

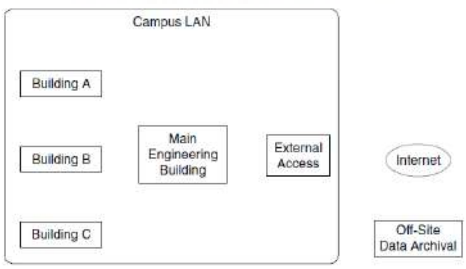
10. What are the primary differences between first-order, second-order, and third-order design products?

PART B

Answer all questions. Each question carries 14 marks. (5 * 14 = 70 Marks)

11. Based on the following application locations, develop an applications map using the template provided

    a. There is a distributed computing application between all compute servers.
    b. There is a data storage/access application between all compute servers and the storage servers in Main Engineering.
    c. There is a data migration application between Main Engineering, External Access Building, and Off-Site Data Archival.



OR

12. A) Service metrics are used to monitor, measure, and verify services in the network and determine if performance requirements are being met. Therefore, service metrics must be meaningful to the network operators, managers, users, and staff. For each performance requirement below, describe a service metric that could be used to measure and verify the requirement.

   d. Reliability of a T1 link between an enterprise network and its ISP, with IP routers on each end of the T1.
   e. Round-trip delay between devices on network A and servers on server LAN B.
   f. Average traffic rate in and out of a compute server, measured across all four of its LAN interfaces over 15-minute intervals.

   B) Given an MTBCF requirement of 8000 hours and an MTTR requirement of 4 hours, calculate an availability requirement.

13. Develop a flow model for real-time/near-real-time flows. How would you characterize the flows for this model? What are likely data sources and sinks? Apply your model to a videoconferencing application.

OR

14. A) A network's architecture differs from its design, in terms of its scope, level of detail, description, and location information. Describe how an architecture and design differ in each characteristic.

   B) Explain the network architectural models based on flows.

15. Given the network address 129.99.0.0/16, develop a variable-length addressing scheme that best fits the design, with the following numbers of users:

AS Number	Location	Department	Users
1	Chicago Campus Building 1	Legal	120
		Accounting	370
	Chicago Campus Building 2	HQ	1580
		Engineering	200
2	Toronto	Sales	75
	Boston	Sales	110
3	Philadelphia	Operations1	2150
		Operations2	975
		Sales	575

OR

16. What are the layers of network management? Give an example of management at each layer (what is managed, how it is managed).

17. For the queuing mechanisms given below, give an example of how each would be used within a network. What problem(s) is each mechanism solving?

    g. First in first out (FIFO)
    h. Class-based queuing (CBQ)
    i. Weighted fair queuing (WFQ)
    j. Random early detect (RED)
    k. Weighted RED (WRED)

OR

18. Explain the process for PPP/PPPoE Session Establishment.

19. Which of the following evaluation criteria most likely apply to equipment evaluations, which ones apply to service-provider evaluations, and which apply to both?

    l. Available service-level agreements (SLAs)
    m. Standards compliance (IETF)
    n. Mean time between failure (MTBF)
    o. Mean time between service outage (MTBSO)
    p. Hardware scalability

OR

20. Using any suitable example, explain criteria refinement and rating development.

Syllabus

Module 1:REQUIREMENT ANALYSIS (8Hours)
Requirements Analysis: objective and background, User Requirements, Application Requirements, Device Requirements, Network Requirements, Other Requirements, Requirements Specification and Map Requirements Analysis: Process- Gathering and Listing Requirements, Developing Service Metrics, Characterizing behaviour, Developing RMA requirements, Developing delay Requirements, Developing capacity Requirements, Environment-Specific Thresholds and Limits, Requirements for Predictable and Guaranteed Performance, Requirements mapping
Module 2 :FLOW ANALYSISAND NETWORK ARCHITECTURE (10 Hours)
Flow Analysis: Background, Flows, Identifying and Developing Flows, Data Sources and Sinks, Flow Models, Flow Prioritization, The Flow Specification, Example Application of Flow Analysis Network Architecture: Background, Component Architectures, Reference Architecture, Architectural Models, Systems and Network Architectures

Module 3: ADDRESSING AND NETWORK MANAGEMENT(10 Hours)
Addressing and Routing Architecture: Background, Addressing Mechanisms, Routing Mechanisms Addressing Strategies, Routing Strategies, Architectural Considerations Network Management Architecture: Background, Network Devices and Characteristics, Network Management Mechanisms, In-Band and Out-of-Band Management, Centralized, Distributed, and Hierarchical Management, Scaling Network Management Traffic
Module 4:PERFORMANCE AND SECURITY ARCHITECTURE (9 Hours)
Performance Architecture: Background, Developing Goals for Performance, Performance Mechanisms, Evaluation of Performance Mechanisms Security and Privacy Architecture: Background, Developing a Security and Privacy Plan, Security and Privacy Administration, Security and Privacy Mechanisms, Evaluation of Security Mechanisms
Module 5: NETWORK DESIGN (8 Hours)
Network Design: Objectives, Design Concepts, Design Process, Vendor, Equipment, and Service-Provider Evaluations, Network Layout, Design Traceability, Design Metrics

Text Books

1. Network Analysis, Architecture and Design, 3RDEdition (The Morgan Kaufmann Series in Networking) James D. McCabe, Elsevier Science (USA), 2007.

Reference Books

1. Computer Networks: A Systems Approach by Larry L. Peterson, Bruce S. Davie - 2007, Elsevier Inc.

2. Top-down Network Design: [a Systems Analysis Approach to Enterprise Network Design] ByPriscilla Oppenheimer, Cisco Press , 3rd Edition, ISBN-13: 978-1-58720-283-4 ISBN-10: 1-58720-283-2

3. Integrated Management of Networked Systems: Concepts, Architectures, and Their Operational Application (The Morgan Kaufmann Series in Networking), Heinz-Gerd Hegering, Sebastian Abeck,and Bernhard Neumair, 1999.

4. “Network Design and Management” – by Steven T.Karris, Orchard publications, Second edition, Copyright 2009, ISBN 978-1-934404-15-7

5. “Network Design, Management and Technical Perspective”, Teresa C. Mann-Rubinson and KornelTerplan, CRC Press, 1999

6. "Ethernet Networks-Design, Implementation, Operation and Management by Gilbert Held,JohnWiley and sons, Fourth Edition

7. James Kurose and Keith Ross, "Computer Networking: A Top-Down Approach Featuring the Internet", 1999

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**REQUIREMENT ANALYSIS**	**8Hours**
1.1	Requirements Analysis: objective and background, User Requirements	1 Hour
1.2	Application Requirements, Device Requirements, Network Requirements, Other Requirements, Requirements Specification and Map	2Hours
1.3	Requirements Analysis: Process- Gathering and Listing Requirements, Developing Service Metrics	1Hour
1.4	Characterizing behaviour, Developing RMA requirements, Developing delay Requirements	2Hours
1.5	Developing capacity Requirements, Environment-Specific Thresholds and Limits, Requirements for Predictable and Guaranteed Performance, Requirements mapping	2 Hours
2	**FLOW ANALYSISAND NETWORK ARCHITECTURE**	**10Hours**
2.1	Flow Analysis: Background, Flows, Identifying and Developing Flows, Data Sources and Sinks	3 Hours
2.2	Flow Models, Flow Prioritization, The Flow Specification, Example Application of Flow Analysis	3Hours
2.3	Network Architecture: Background, Component Architectures	1 Hour
2.3	Reference Architecture,Architectural Models, Systems and Network Architectures	3 Hours
3	**ADDRESSING AND NETWORK MANAGEMENT**	**10 Hours**
3.1	Addressing and Routing Architecture: Background, Addressing Mechanisms	3 Hours
3.2	Routing Mechanisms Addressing Strategies, Routing Strategies, Architectural Considerations	3 Hours
3.3	Network Management Architecture: Background, Network Devices and Characteristics, Network Management Mechanisms	2 Hours
3.4	In-Band and Out-of-Band Management, Centralized, Distributed, and Hierarchical Management, Scaling Network Management Traffic	2 Hours
4	**PERFORMANCE AND SECURITY ARCHITECTURE**	**9Hours**
4.1	Performance Architecture: Background, Developing Goals for Performance	1 Hour
4.2	Performance Mechanisms, Evaluation of Performance	2Hours

	Mechanisms	
4.3	Security and Privacy Architecture: Background, Developing a Security and Privacy Plan	2Hours
4.4	Security and Privacy Administration	2 Hours
4.5	Security and Privacy Mechanisms, Evaluation of Security Mechanisms	2 Hours
5	**NETWORK DESIGN**	**8Hours**
5.1	Network Design: Objectives, Design Concepts	2Hours
5.2	Design Process, Vendor, Equipment, and Service-Provider Evaluations	3Hours
5.3	Network Layout, Design Traceability, Design Metrics	3Hours

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT396	APPLIED COMPUTER SECURITY	VAC	3	1	0	4

Preamble: The syllabus is designed with the view of preparing the students capable of understanding various application areas of computer security. The students will get to know about vulnerabilities and threats occurring in various information systems. Furthermore, the student will be able to identify suitable security mechanisms to handle security issues faced by various applications.

Prerequisite: ITT206 Database management systems , ITT303 Operating systems, ITT305 Data communication and Networking, ITT395 Security in Computing,

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome(CO)	Bloom's Category Level
CO 1	Outline the basic concepts and techniques to secure various applications	Level 2: Understand
CO 2	Explain various security mechanisms for storage infrastructure	Level 2: Understand
CO 3	Interpret the security techniques of IoT environment	Level 2: Understand
CO 4	Apply the underlying principles of security to cloud computing platforms	Level 3: Apply
CO 5	Apply the significance of blockchain technology for securing various applications	Level 3: Apply

Mapping of course outcomes with program outcomes

POs / COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO 1	3	2	2	1	1	1	1	1	-	-	-	2
CO 2	3	2	2	1	1	-	-	1	-	-	-	2
CO 3	3	2	2	1	1	1	-	1	-	-	-	1
CO 4	3	2	1	1	2	-	-	2	-	-	-	2
CO 5	3	3	2	2	2	2	1	2	2	1	-	2

3/2/1: high/medium/low

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	Test1(Marks)	Test2(Marks)	
Remember	10	10	20
Understand	40	20	40
Apply		20	40
Analyze			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Explain various issues and challenges in smartphone security.
2. Summarize key success factors for biometric systems.
3. Outline various countermeasures to protect from threats posed through emails.

Course Outcome 2 (CO2):

1. Compare auditing the security of a storage environment with SAN, NAS, and iSCSI implementations.
2. Outline various threats and vulnerabilities occur in virtualized and cloud environments.
3. Explain various security concerns and measures in the virtualized and cloud environment.

Course Outcome 3 (CO3):

1. Summarize various attack types that relate to IoT
2. Outline the importance of IoT security CONOPS document in IoT System development.
3. Explain the IoT System security verification and validation (V&V) procedure.

Course Outcome 4 (CO4):

1. Identify the effect of six surfaces of attacks in a cloud computing environment on SaaS,PaaS,Iaas.
2. Compare the benefits and the potential problems due to virtualization on public,private and hybrid clouds.
3. Identify the implications of a two-level security model of commodity operating systems.

Course Outcome 5 (CO5):

1. Explain various building blocks of blockchain technology.
2. Identify the ways in which blockchain technology can be compromised.
3. Apply blockchain technology to solve automotive security and privacy issues.

Model Question Paper

Course Code: ITT396

Course Name: APPLIED COMPUTER SECURITY

Max.Marks:100 Duration: 3 Hour

PART A

(Each Question carries 3 Marks) (10*3=30)

1. Explain security threats associated with wearable devices.
2. Outline various security issues and concerns in electronic commerce.
3. Explain the concept of risk triad.
4. Outline IP SAN security implementation in storage networking.
5. Summarize the essential components of information assurance (IA).
6. Compare the use of attack trees with that of an attack vector for understanding a system's security posture.
7. Examine the security risks posed by the cloud users.
8. Identify the Surfaces of attacks in a cloud computing environment.
9. Design the structure of a block in a blockchain with the help of a neat diagram.

10. Obtain blockchain and its operations.

PART B

(Answer all Question. Each question carries 14 Marks) (5*14=70)

11. Explain key success factors for biometric systems.

OR

12. Summarize various security threats posed by electronic mails and its countermeasures.

13. Illustrate various storage security domains with suitable diagrams.

OR

14. Outline security implementations in storage networking. (i) FC SAN and (ii) NAS

15. a. Explain the need for building security into design and development. (7)

    b. Summarize the issues and techniques related to securely engineering IoT systems. (7)

OR

16. a. Explain the implementation and integration aspects of the IoT security lifecycle. (10)

    b. Outline the IoT System security verification and validation (V&V) procedure. (4)

17. a. Identify the indications of the lack of trusted paths in commodity operating systems with examples showing the effects of this deficiency. (10)

    b. Obtain the two-level security model of commodity operating systems. (4)

OR

18. a.Obtain the Security risks posed by a management OS. (9)

    b. Identify the privacy concerns for three cloud delivery models. (5)

19. Identify various peer to peer attacks on blockchain and its countermeasures.

OR

20. a. Identify various application oriented attacks on blockchain and its countermeasures. (7)

    b. Apply blockchain technology to solve security and privacy issues in various applications. Explain with an example. (7)

Syllabus

Module 1: Security of Applications (10 Hours)
Security Considerations in Mobile and Wireless Computing, Smartphone Security, Biometrics for Security, Security of Electronic Mail Systems, Security of Electronic Commerce. (T1: Chapter - 6, 8, 11,18 & 19)
Module 2: Storage Security (8 Hours)
Securing the storage Infrastructure - Information Security Framework, Risk Triad, Storage Security Domains, Security Implementation in Storage Networking, Securing Storage Infrastructure in virtualized and Cloud Environments, RSA and VMware Security Products. (T2: Chapter - 14)
Module 3 : IOT Security (9 Hours)
IoT and cyber-physical systems, IoT security (vulnerabilities, attacks, and countermeasures)-Common IoT attack types-Attack trees-Building an attack tree, security engineering for IoT development-Secure design-Safety and security design, IoT security lifecycle-The secure IoT system implementation lifecycle -Implementation and integration-IoT security CONOPS document-Network and security integration-System security verification and validation (V&V)-Security training-Secure configurations. (T3: Chapter - 2,3 & 4)
Module 4 : Cloud Security (8 Hours)
Introduction to Cloud and Virtualization- Cloud Security - Cloud Security Risks, Trust, Operating System Security, VM Security, Security of Virtualization, Security Risks Posed by Shared Images, Security Risks Posted by Management OS. (T4: Chapter -9)
Module 5 : Distributed Systems Security (10 Hours)
Introduction to Blockchain - Blockchain Overview, Blockchain Building Blocks, Blockchain Commercial Use cases, Blockchain Military Cyber Operations Use cases, Blockchain Challenges, Overview of Attack Surfaces in Blockchain - Overview of Blockchain and its Operations, Blockchain Attacks, Blockchain's Peer to Peer System, Application Oriented Attacks, A Blockchain-based Solution to Automotive Security and Privacy - The Proposed Framework, Applications. (T5: Chapter - 1, 3 & 5)

Text Books

T1. Nina Godbole, "Information System Security", 2nd Edition, Wiley

T2. Information Storage and Management: Storing, Managing, and Protecting Digital Information in Classic, Virtualized, and Cloud Environments, Wiley; 2nd Edition, EMC Corporation, 2012.

T3. B.Rusell and D. Van Duren, "Practical Internet of Things Security," Packt Publishing, 2016.

T4. Marinescu D C, Cloud Computing Theory and Practice, Morgan Kaufmann (2014).

T5. Sachin Shetty, Charles A. Kamhoua , Laurent L. Njilla , "Blockchain for Distributed Systems Security", Wiley

Reference Books

R1: Sarika Gupta, "Information and Cyber Security", Khanna Publishing House, Delhi.
R2: Zhou, Honbo. The internet of things in the cloud: A middleware perspective. CRC press, 2012.
R3: Antony Lewis, "The Basics of Bitcoins and Blockchains", Mango Publishing

Course Contents and Lecture Schedule

1	**Security of Applications**	**10** Hours
1.1	Security Considerations in Mobile and Wireless Computing	2 Hours
1.2	Smartphone Security	2 Hours
1.3	Biometrics for Security	2 Hours
1.4	Security of Electronic Mail Systems	2 Hours
1.5	Security of Electronic Commerce	2 Hours
2	**Storage Security**	**8 Hours**
2.1	Securing the storage Infrastructure - Information Security Framework,	1 Hour
2.2	Risk Triad	1 Hour
2.3	Storage Security Domains	2 Hours
2.4	Security Implementation in Storage Networking	2 Hours
2.5	Securing Storage Infrastructure in virtualized and Cloud Environments	1 Hour
2.6	RSA and VMware Security Products.	1 Hour
3	**IoT Security**	**9 hrs**
3.1	Introduction: IoT and cyber-physical systems.	1 Hour
3.2	IoT security (vulnerabilities, attacks, and countermeasures)- Common IoT attack types-Attack trees-Building an attack tree.	2 Hours
3.3	Security engineering for IoT development-Secure design-Safety and security design.	1 Hour
3.4	IoT security lifecycle-The secure IoT system implementation lifecycle - Implementation and integration-IoT security CONOPS document.	2 Hours
3.5	Network and security integration.	1 Hour

3.6	System security verification and validation (V&V)	1 Hour
3.7	Security training-Secure configurations	1 Hour
4	**Cloud Security**	**8 hrs**
4.1	Introduction to Cloud and Virtualization, Cloud Security - Cloud Security Risks-Security: The top concern for cloud users-Privacy and privacy impact assessment	2 Hours
4.2	Trust, Operating System Security	1 Hour
4.3	VM Security	1 Hour
4.4	Security of Virtualization	2 Hours
4.5	Security Risks Posed by Shared Images	1 Hour
4.6	Security Risks Posted by Management OS	1 Hour
5	**Distributed Systems Security**	**10 hrs**
5.1	Introduction to Distributed systems and Blockchain - Blockchain Overview, Blockchain Building Blocks	2 Hours
5.2	Blockchain Commercial Use cases	1 Hour
5.3	Blockchain Military Cyber Operations Use cases, Blockchain Challenges	1 Hour
5.4	Overview of Attack Surfaces in Blockchain - Overview of Blockchain and its Operations	1 Hour
5.5	Blockchain Attacks	1 Hour
5.6	Blockchain's Peer to Peer System	1 Hour
5.7	Application Oriented Attacks	1 Hour
5.8	A Blockchain-based Solution to Automotive Security and Privacy	1 Hour
5.9	The Proposed Framework, Applications.	1 Hour

CODE	COURSE NAME	CATEGORY	L	T	P	CREDIT
ITT398	EMBEDDED SYSTEM	VAC	3	1	0	4

Preamble: Embedded Systems course is intended to deliver students the concepts of embedded hardware and software. It also helps them to develop embedded systems using Raspberry Pi and Arduino.

Prerequisites:

- Basics of Computer programming,
- ITT296 Microprocessor and Microcontroller programming,
- ITT397 Advanced Computer Architecture

Course Outcomes: After the completion of the course the student will be able to

CO No.	Course Outcome (CO)	Bloom's Category Level
CO1	Discuss the concepts of embedded systems, micro controllers and sensors	Level2: Understand
CO2	Describe examples of embedded systems, buses, protocol and ARM Processors	Level 2: Understand
CO3	Implement embedded system design techniques and software development tools	Level 3: Apply
CO4	Apply the basic concepts of real time operating systems using Raspberry Pi.	Level 3: Apply
CO5	Develop small or medium scale embedded systems using Arduino	Level 3: Apply

Mapping of course outcomes with program outcomes

POs / COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	1	-	-	2	1	-	-	-	-	-
CO2	3	3	3	-	2	2	2	-	-	-	-	2
CO3	3	3	3	-	2	2	1	-	-	-	-	2
CO4	3	3	3	-	3	2	2	-	3	2	2	3
CO5	3	3	3	-	3	3	2	-	3	2	2	3

3/2/1: High/Medium/Low

Assessment Pattern

Bloom's Category	Continuous Assessment Test		End Semester Examination
	Test 1(Marks)	Test 2(Marks)	
Remember	15	15	30
Understand	30	15	30
Apply	5	20	40
Analyse			
Evaluate			
Create			

Mark distribution

Total Marks	CIE	ESE	ESE Duration
150	50	100	3 hours

Continuous Internal Evaluation Pattern:

Attendance : 10 marks
Continuous Assessment Test (2 numbers) : 25 marks
Assignment/Quiz/Course project : 15 marks

End Semester Examination Pattern: There will be two parts; Part A and Part B. Part A contain 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which student should answer any one. Each question can have maximum 2 sub-divisions and carry 14 marks.

Sample Course Level Assessment Questions

Course Outcome 1 (CO1):
1. With the help of a diagram explain the generic model of an embedded system.
2. Working principle of sensors.

Course Outcome 2 (CO2):
1. Describe examples of embedded systems.
2. Explain buses and protocols in Embedded systems.

Course Outcome 3 (CO3):
1. Describe the importance of design techniques in the development of embedded systems.
2. How software development tools helps in embedded programming.
3. Consider an embedded system-GPS moving map that displays the map of a terrain around the

users' current position. List the various requirements to design GPS moving map and prepare a requirement form or chart.

Course Outcome 4 (CO4):
1. With an example explain real time scheduling algorithm.
2. Write a program to illustrate reading a button using PI.

Course Outcome 5 (CO5):
1. With an example illustrate Read/write operation for Digital and Analog Input/output.
2. Write an Arduino Sketch to Read/write operation from a digital camera.

Model Question Paper

Course Code: ITT398
Course Name: EMBEDDED SYSTEM
Max.Marks:100 **Duration: 3 Hour**

PART A
(Each question carries 3 Marks) (10*3=30)

1.Explain the features of embedded system.
2.Briefly discuss about the working principle of IR sensor.
3.Define buses and protocol.
4.List any 3 applications of embedded system.
5.Why it is said that a requirement bug is more costlier than a coding bug?
6.Explain the difference between emulator and simulator.
7. Write example for hard, soft and firm real time tasks.
8.Differentiate Arduino and Raspberry Pi.
9.Make a program that light up LED in any pattern.
10.Write an Arduino sketch to read a RF Id tag

PART B
(Each full question carries 14 marks) (5*14=70)

11. a) What are the merits of embedded systems? (7 marks)

b) Distinguish control interface and data interface for an ADC. (7 marks)

OR

12. a) Explain the working of range and temperature sensor. (8 marks)

b) What are the factors to be considered for low power design of embedded systems? (6 marks)

13. a) The content of some registers are given as: R1=0xEF00DE12, R2=0x0456123F, R5=4, r6=28 Find the result when following instructions are executed?

i) LSL R1,#8

ii) ASR R1,R5

iii) ROR R2,R6

iv) LSR R2,#5 (8 marks)

b) With a neat diagram explain fundamental blocks of a mobile phone and explain the function of each block (6 marks)

OR

14. a) With diagram explain Following buses

i) SPI

ii) USB. (8 marks)

b) Explain ARM Processor's Memory Organization with neat diagram. (6 marks)

15. a) Consider an embedded system-GPS moving map that displays the map of a terrain around the users' current position. List the various requirements to design GPS moving map and prepare a requirement form or chart. (7 marks)

b) Explain the embedded program development environment. (7 marks)

OR

16. a) Discuss how quality is assured by verifying specification. (7 marks)

b) Discuss Software Development Tools used in embedded systems. (7 marks)

17. a) Explain Inter process Communication Mechanisms. With example. (7 marks)

b) What are the general characteristics that make Raspberry Pi become popular in market. (7 marks)

OR

18. a) Explain about Real time scheduling algorithms. (7 marks)

b) Design a lamp timer using Raspberry Pi and schedule the timer using Cron utility such that lamp is switched ON at 10pm and Switched OFF at 5pm only for weekends . (7 marks)

19. a) Write an Arduino sketch to display only EVEN numbers using 7 segments display in common anode connection. (7 marks)

b) A manufacturer wants to design a new safety feature into their cars. The safety feature simply alerts the driver if the speed of the car exceeds 100 km/h by beeping a buzzer constantly until the speed drops below 100 km/h again. Develop a project, suitable for an Arduino that implements the functionality required by the manufacturer. Assume that the speedometer of the car is connected to ADC pin 5 of the Arduino and the buzzer is connected to digital pin 13. The range of the speedometer is from 0 km/h to 140km/h where:

- 0 km/h represents an ADC value of 0
- 140 km/h represents an ADC value of 1023

. (7 marks)

OR

20. a) Write an Arduino program for Getting Location from a GPS. (7 marks)

b) Write an Arduino program that light up LED from top to bottom and then goes backward with only one LED is ON at any time. (7 marks)

Syllabus

Module 1: Embedded Systems-Introduction (10 hours)
Embedded Systems-Introduction, Features, Model, Merits, Classification. The Hardware Point of View- Microcontroller Unit, 8 bit MCU, Memory for Embedded System-Flash memory, Low Power Design Sensors, ADCs and Actuators-Temperature Sensors, Light Sensors, Range Sensors, Humidity Sensors, Other Sensors, Analog to Digital Converters, Actuators.
Module 2: ARM Processor,Buses and Protocols (10 hours)
ARM Processor- Processor and Memory Organization, Data Operations, Flow of Control Buses and Protocols – Defining Buses and Protocols, On-board buses for Embedded Systems, External Buses, Automotive Buses Examples of Embedded Systems –Automotive Electronics, RFID, Robotics, Brain Machine Interface
Module 3: Embedded Systems Design Techniques and Software Development Tools (7 hours)
Embedded Systems Design Techniques – Design Methodologies, Requirements Analysis, Specifications, System Analysis and Architecture Design, Quality Assurance, Design Examples Software Development Tools-Embedded program development, Downloading the hexfile to Non-volatile memory- Emulator- Hardware simulator.
Module 4: RTOS and Raspberry Pi (9 hours)
Real Time Operating Systems, Real time Scheduling algorithms-RM,EDF Inter process Communication Mechanisms Raspberry Pi – Introduction, Python and Raspberry Pi, Arduino and Raspberry Pi, Basic Input and Output, Programming Inputs and Outputs with Python
Module 5: Interfacing of micro-controllers (9 hours)
Arduino – Introduction-Making the Sketch Simple Digital and Analog Input-Using a Switch, Reading a Keypad, Reading Analog Values Getting Input from Sensors-Detecting Movement, light, motion, distance, vibration, sound and temperature Reading RFID Tags, Getting Location from a GPS Visual Output-Connecting and Using LEDs Physical Output-Controlling the Position of a Servo Remotely Controlling External Devices-Controlling a Digital Camera

Text Books

1. Embedded Systems : An Integrated Approach -Lyla B Das, Pearson Education, 2013
2. Getting Started With Raspberry Pi-Matt Richardson, Shawn Wallace, O'Reilly, 2013
3. Arduino Cookbook -Michael Margolis, O'Reilly, 2011
4. Modern Embedded Computing -Peter Barry, Patrick Crowley, Morgan Kaufmann 2012
5. Computers as Components : Principles of Embedded Computing System Design -Wayne Wolf, Elsevier

Reference Books

1. Embedded Systems- Architecture Programming and design -Raj Kamal , McGraw-Hill, second edition
2. Introduction to Arduino Alan G. Smith , CreateSpace Independent Publishing Platform , 2011

Course Contents and Lecture Schedule

No	Topic	No. of Lectures
1	**Embedded Systems-Introduction**	**10 Hours**
1.1	Embedded Systems-Introduction, Features	1 Hour
1.2	Embedded Systems-Model, Merits, Classification.	1 Hour
1.3	The Hardware Point of View- Microcontroller Unit,	1 Hour
1.4	Memory for Embedded System-Flash memory	1 Hour
1.5	8 bit MCU	1 Hour
1.6	Low Power Design	1 Hour
1.7	Sensors -Temperature Sensors, Light Sensors	1 Hour
1.8	Sensors-Range Sensors, Humidity Sensors, Other Sensors,	1 Hour
1.9	Actuators	1 Hour
1.10	Analog to Digital Converters	1 Hour
2	**ARM Processor,Buses and Protocols**	**10 Hours**
2.1	ARM Processor- Processor and Memory Organization	1 Hour
2.2	ARM-Data Operations	1 Hour
2.3	ARM- Flow of Control	1 Hour

2.4	Buses and Protocols – Defining Buses and Protocols	1 Hour
2.5	On-board buses for Embedded Systems	1 Hour
2.6	External Buses	1 Hour
2.7	Automotive Buses	1 Hour
2.8	Examples of Embedded Systems –Automotive Electronics	1 Hour
2.9	RFID, Robotics	1 Hour
2.10	Brain Machine Interface	1 Hour
3	**Embedded Systems Design Techniques and Software Development Tools**	**7 Hours**
3.1	Embedded Systems Design Techniques	1 Hour
3.2	Design Methodologies, Requirements Analysis and Architecture	1 Hour
3.3	Specifications, System Analysis	1 Hour
3.4	Quality Assurance, Design Examples	1 Hour
3.5	Software Development Tools-Embedded program development	1 Hour
3.6	Downloading the hexfile to Non-volatile memory	1 Hour
3.7	Emulator- Hardware simulator	1 Hour
4	**RTOS and Raspberry Pi**	**9 Hours**
4.1	Real Time Operating Systems	1 Hour
4.2	Real time Scheduling algorithms-RM	1 Hour
4.3	Real time Scheduling algorithms-EDF	2 Hours
4.4	Inter process Communication Mechanisms	2 Hours
4.5	Raspberry Pi – Introduction	1 Hour
4.6	Python and Raspberry Pi, Arduino and Raspberry Pi	1 Hour
4.7	Basic Input and Output, Programming Inputs and Outputs with Python	1 Hour
5	**Interfacing of micro-controllers**	**9 Hours**
5.1	Arduino – Introduction-Making the Sketch	1 Hour
5.2	Simple Digital and Analog Input-Using a Switch	2 Hours
5.3	Reading a Keypad, Reading Analog Values	1 Hour
5.5	Reading RFID Tags, Getting Location from a GPS	2 Hours
5.6	Visual Output-Connecting and Using LEDs	1 Hour
5.7	Physical Output-Controlling the Position of a Servo	1 Hour
5.8	Remotely Controlling External Devices-Controlling a Digital	1 Hour

ITD415	PROJECT PHASE I	CATEGORY	L	T	P	CREDIT
		PWS	0	0	6	2

Preamble: The course 'Project Work' is mainly intended to evoke the innovation and invention skills in a student. The course will provide an opportunity to synthesize and apply the knowledge and analytical skills learned, to be developed as a prototype or simulation. The project extends to 2 semesters and will be evaluated in the 7^{th} and 8^{th} semester separately, based on the achieved objectives. One third of the project credits shall be completed in 7^{th} semester and two third in 8^{th} semester. It is recommended that the projects may be finalized in the thrust areas of the respective engineering stream or as interdisciplinary projects. Importance should be given to address societal problems and developing indigenous technologies.

Course Objectives

- To apply engineering knowledge in practical problem solving.
- To foster innovation in design of products, processes or systems.
- To develop creative thinking in finding viable solutions to engineering problems.

Course Outcomes [COs] : After successful completion of the course, the students will be able to:

CO1	Model and solve real world problems by applying knowledge across domains (Cognitive knowledge level: **Apply**).
CO2	Develop products, processes or technologies for sustainable and socially relevant applications (Cognitive knowledge level: **Apply**).
CO3	Function effectively as an individual and as a leader in diverse teams and to comprehend and execute designated tasks (Cognitive knowledge level: **Apply**).
CO4	Plan and execute tasks utilizing available resources within timelines, following ethical and professional norms (Cognitive knowledge level: **Apply**).
CO5	Identify technology/research gaps and propose innovative/creative solutions (Cognitive knowledge level: **Analyze**).
CO6	Organize and communicate technical and scientific findings effectively in written and oral forms (Cognitive knowledge level: **Apply**).

Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	2	1	2	2	2	1	1	1	1	2
CO2	2	2	2		1	3	3	1	1		1	1
CO3									3	2	2	1
CO4					2			3	2	2	3	2
CO5	2	3	3	1	2							1
CO6					2			2	2	3	1	1

Abstract POs defined by National Board of Accreditation			
PO#	**Broad PO**	**PO#**	**Broad PO**
PO1	Engineering Knowledge	PO7	Environment and Sustainability
PO2	Problem Analysis	PO8	Ethics
PO3	Design/Development of solutions	PO9	Individual and team work
PO4	Conduct investigations of complex problems	PO10	Communication
PO5	Modern tool usage	PO11	Project Management and Finance
PO6	The Engineer and Society	PO12	Lifelong learning

PROJECT PHASE I

Phase 1 Target

- Literature study/survey of published literature on the assigned topic
- Formulation of objectives
- Formulation of hypothesis/ design/ methodology
- Formulation of work plan and task allocation.
- Block level design documentation
- Seeking project funds from various agencies
- Preliminary Analysis/Modeling/Simulation/Experiment/Design/Feasibility study
- Preparation of Phase 1 report

Evaluation Guidelines & Rubrics

Total: 100 marks (Minimum required to pass: 50 marks).

- Project progress evaluation by guide: 30 Marks.
- Interim evaluation by the Evaluation Committee: 20 Marks.
- Final Evaluation by the Evaluation Committee: 30 Marks.
- Project Phase - I Report (By Evaluation Committee): 20 Marks.

(The evaluation committee comprises HoD or a senior faculty member, Project coordinator and project supervisor).

Evaluation by the Guide

The guide/supervisor shall monitor the progress being carried out by the project groups on a regular basis. In case it is found that progress is unsatisfactory it shall be reported to the Department Evaluation Committee for necessary action. The presence of each student in the group and their involvement in all stages of execution of the project shall be ensured by the guide. Project evaluation by the guide: 30 Marks. This mark shall be awarded to the students in his/her group by considering the following aspects:

Topic Selection: innovativeness, social relevance etc. (2)

Problem definition: Identification of the social, environmental and ethical issues of the project problem. (2)

Purpose and need of the project: Detailed and extensive explanation of the purpose and need of the project. (3)

Project Objectives: All objectives of the proposed work are well defined; Steps to be followed to solve the defined problem are clearly specified. (2)

Project Scheduling & Distribution of Work among Team members: Detailed and extensive Scheduling with timelines provided for each phase of project. Work breakdown structure well defined. (3)

Literature survey: Outstanding investigation in all aspects. (4)

Student's Diary/ Daily Log: The main purpose of writing daily diary is to cultivate the habit of documenting and to encourage the students to search for details. It develops the students' thought process and reasoning abilities. The students should record in the daily/weekly activity diary the day to day account of the observations, impressions, information gathered and suggestions given, if any. It should contain the sketches & drawings related to the observations made by the students. The daily/weekly activity diary shall be signed after every day/week by the guide. (7)

Individual Contribution: The contribution of each student at various stages. (7)

EVALUATION RUBRICS for PROJECT Phase I: Interim Evaluation

No.	Parameters	Marks	Poor	Fair	Very Good	Outstanding
1-a	Topic identification, selection, formulation of objectives and/or literature survey. (Group assessment) [CO1]	10	The team has failed to come with a relevant topic in time. Needed full assistance to find a topic from the guide. They do not respond to suggestions from the evaluation committee and/or the guide. No literature review was conducted. The team tried to gather easy information without verifying the authenticity. No objectives formed yet.	The team has identified a topic. The originally selected topic lacks substance and needs to be revised. There were suggestions given to improve the relevance and quality of the project topic. Only a few relevant references were consulted/ studied and there is no clear evidence to show the team's understanding on the same. Some objectives identified, but not clear enough.	Good evidence of the group thinking and brainstorming on what they are going to build. The results of the brainstorming are documented and the selection of topic is relevant. The review of related references was good, but there is scope of improvement. Objectives formed with good clarity, however some objectives are not realistic enough.	The group has brainstormed in an excellent manner on what they were going to build. The topic selected is highly relevant, real world problem and is potentially innovative. The group shows extreme interest in the topic and has conducted extensive literature survey in connection with the topic. The team has come up with clear objectives which are feasible.
			(0 – 3 Marks)	(4 – 6 Marks)	(7 - 9 Marks)	(10 Marks)
1-b	Project Planning, Scheduling and Resource/ Tasks Identification and allocation. (Group assessment) [CO4]	10	No evidence of planning or scheduling of the project. The students did not plan what they were going to build or plan on what materials / resources to use in the project. The students do not have any idea on the budget required. The team has not yet decided on who does what. No project journal kept.	Some evidence of a primary plan. There were some ideas on the materials /resources required, but not really thought out. The students have some idea on the finances required, but they have not formalized a budget plan. Schedules were not prepared. The project journal has no details. Some evidence on task allocation among the team members.	Good evidence of planning done. Materials were listed and thought out, but the plan wasn't quite complete. Schedules were prepared, but not detailed, and needs improvement. Project journal is presented but it is not complete in all respect / detailed. There is better task allocation and individual members understand about their tasks. There is room for improvement.	Excellent evidence of enterprising and extensive project planning. Gantt charts were used to depict detailed project scheduling. A project management/version control tool is used to track the project, which shows familiarity with modern tools. All materials / resources were identified and listed and anticipation of procuring time is done. Detailed budgeting is done. All tasks were identified and incorporated in the schedule. A well-kept project journal shows evidence for all the above, in addition to the interaction with the project guide. Each member knows well about their individual tasks.
			(0 – 3 Marks)	(4 – 6 Marks)	(7 - 9 Marks)	(10 Marks)

Phase 1 Interim Evaluation Total Marks: 20

EVALUATION RUBRICS for PROJECT Phase I: Final Evaluation

Sl. No.	Parameters	Marks	Poor	Fair	Very Good	Outstanding
1-c	Formulation of Design and/or Methodology and Progress. (Group assessment) [CO1]	5	None of the team members show any evidence of knowledge about the design and the methodology adopted till now/ to be adopted in the later stages. The team has not progressed from the previous stage of evaluation.	The students have some knowledge on the design procedure to be adopted, and the methodologies. However, the team has not made much progress in the design, and yet to catch up with the project plan.	The students are comfortable with design methods adopted, and they have made some progress as per the plan. The methodologies are understood to a large extent.	Shows clear evidence of having a well- defined design methodology and adherence to it. Excellent knowledge in design procedure and its adaptation. Adherence to project plan is commendable.
			(0 – 1 Marks)	(2 – 3 Marks)	(4 Marks)	(5 Marks)
1-d	Individual and Teamwork Leadership (Individual assessment) [CO3]	10	The student does not show any interest in the project activities, and is a passive member.	The student show some interest and participates in some of the activities. However, the activities are mostly easy and superficial in nature.	The student shows very good interest in project, and takes up tasks and attempts to complete them. Shows excellent responsibility and team skills. Supports the other members well.	The student takes a leadership position and supports the other team members and leads the project. Shows clear evidence of leadership.
			(0 – 3 Marks)	(4 – 6 Marks)	(7 - 9 Marks)	(10 Marks)
1-e	Preliminary Analysis/ Modeling / Simulation/ Experiment / Design/ Feasibility study [CO1]	10	The team has not done any preliminary work with respect to the analysis/modeling/ simulation/experiment/design/feasibility study/ algorithm development.	The team has started doing some preliminary work with respect to the project. The students however are not prepared enough for the work and they need to improve a lot.	There is some evidence to show that the team has done good amount of preliminary investigation and design/ analysis/ modeling etc. They can improve further.	Strong evidence for excellent progress in the project. The team has completed the required preliminary work already and are poised to finish the phase I in an excellent manner. They have shown results to prove their progress.
			(0 – 3 Marks)	(4 – 6 Marks)	(7 - 9 Marks)	(10 Marks)

1-f	Documentation and presentation. (Individual & group assessment). [CO6]	5	The team did not document the work at all. The project journal/diary is not presented. The presentation was shallow in content and dull in appearance. The individual student has no idea on the presentation of his/her part.	Some documentation is done, but not extensive. Interaction with the guide is minimal. Presentation include some points of interest, but overall quality needs to be improved. Individual performance to be improved.	Most of the project details were documented well enough. There is scope for improvement. The presentation is satisfactory. Individual performance is good.	The project stages are extensively documented in the report. Professional documentation tools like LaTeX were used to document the progress of the project along with the project journal. The documentation structure is well-planned and can easily grow into the project report. The presentation is done professionally and with great clarity. The individual's performance is excellent.
			(0 – 1 Marks)	(2 – 3 Marks)	(4 Marks)	(5 Marks)
Total		**30**	**Phase - I Final Evaluation Marks: 30**			

EVALUATION RUBRICS for PROJECT Phase I: Report Evaluation						
Sl. No.	**Parameters**	**Marks**	**Poor**	**Fair**	**Very Good**	**Outstanding**
1-g	Report [CO6]	20	The prepared report is shallow and not as per standard format. It does not follow proper organization. Contains mostly Unacknowledged content. Lack of effort in preparation is evident.	Project report follows the standard format to some extent. However, its organization is not very good. Language needs to be improved. All references are not cited properly in the report.	Project report shows evidence of systematic documentation. Report is following the standard format and there are only a few issues. Organization of the report is good. Most of references are cited properly.	The report is exceptionally good. Neatly organized. All references cited properly. Diagrams/Figures, Tables and equations are properly numbered, and listed and clearly shown. Language is excellent and follows standard styles.
			(0 - 7 Marks)	(8 - 12 Marks)	(13 - 19 Marks)	(20 Marks)
Phase - I Project Report Marks: 20						