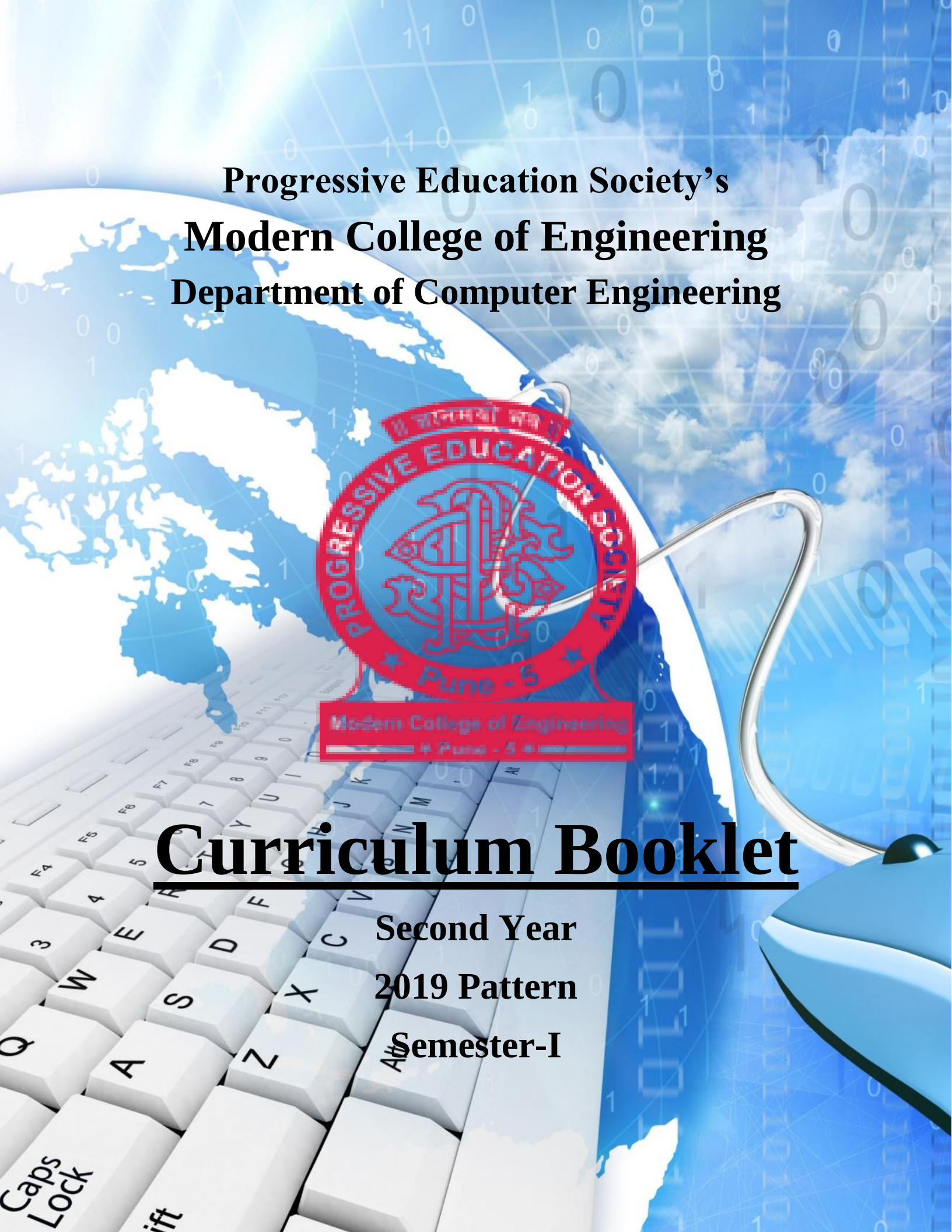


Progressive Education Society's

Modern College of Engineering

Department of Computer Engineering

Curriculum Booklet

Second Year

2019 Pattern

Semester-I

Progressive Education Society's

Modern College of Engineering, Shivajinagar, Pune-05.

Department of Computer Engineering

Curriculum Booklet

2019 Pattern

Semester: I

Class: SE (COMPUTER ENGINEERING)

Vision of the Institute:

- Creation of a collaborative academic environment to foster professional excellence and ethical values.

Mission of the Institute:

- To develop outstanding professionals with high ethical standards capable of creating and managing global enterprises.
- To foster innovation and research by providing a stimulating learning environment.
- To ensure equitable development of students of all ability levels and backgrounds.
- To be responsive to changes in technology, socio-economic and environmental conditions.
- To foster and maintain mutually beneficial partnerships with alumni and industry.

Vision of the Department

- To achieve excellence in the field of computing through quality education.

Mission of the Department

- To develop promising professionals in the field of computing.
- To provide exposure to emerging technologies and inculcate ethics.
- To strengthen association with alumni and industry.

Objectives of the Institute:

- To develop infrastructure appropriate for delivering quality education
- To develop the overall personality of students who will be innovators and future leaders capable of prospering in their work environment.
- To inculcate ethical standards and make students aware of their social responsibilities.
- Promote close interaction among industry, faculty and students to enrich the learning process and enhance career opportunities.
- Encourage faculty in continuous professional growth through quality enhancement programs and research and development activities.

Foster a healthy work environment which allows for freedom of expression and protection of the rights of all stakeholders through open channels of communication

PEO

The graduates of Computer Engineering Department will be,

PEO1: Capable of solving real world problems.

PEO2: Capable of working with multidisciplinary projects.

PEO3: Capable to adapt to changing technologies and life management skills.

PEO4: Able to exhibit professional and ethical responsibilities.

Program Specific Outcomes

Graduate of computer engineering programme will demonstrate

- The ability to understand, analyze, develop and evaluate system based on various algorithmic approaches.
- The ability to pursue career in IT industries, to become an entrepreneur and have zest for higher studies.
- The ability to solve problems using engineering principles, tools and techniques.

Program Outcome

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Course Structure

Curriculum for Second Year of Computer Engineering (2019 Course), Savitribai Phule Pune University

Savitribai Phule Pune University
Second Year of Computer Engineering (2019 Course)
(With effect from Academic Year 2020-21)

Semester-III

Course Code	Course Name	Teaching Scheme (Hours/Week)			Examination Scheme and Marks						Credit Scheme			
		Lecture	Practical	Tutorial	Mid-Sem	End-Sem	Term work	Practical	Oral	Total	Lecture	Practical	Tutorial	Total
210241	Discrete Mathematics	03	-	-	30	70	-	-	-	100	03	--	-	03
210242	Fundamentals of Data Structures	03	-	-	30	70	-	-	-	100	03	-	-	03
210243	Object Oriented Programming (OOP)	03	-	-	30	70	-	-	-	100	03	-	-	03
210244	Computer Graphics	03	-	-	30	70	-	-	-	100	03	-	-	03
210245	Digital Electronics and Logic Design	03	-	-	30	70	-	-	-	100	03	-	-	03
210246	Data Structures Laboratory	-	04	-	-	-	25	50	-	75	-	02	-	02
210247	OOP and Computer Graphics Laboratory	-	04	-	-	-	25	25	-	50	-	02	-	02
210248	Digital Electronics Laboratory	-	02	-	-	-	25	-	-	25	-	01	-	01
210249	Business Communication Skills	-	02	-	-	-	25	-	-	25	-	01	-	01
210250	Humanity and Social Science	-	-	01	-	-	25	-	-	25	-	-	01	01
210251	Audit Course 3													
										Total Credit	15	06	01	22
	Total	15	12	01	150	350	125	75	-	700	-	-	-	-

Subject 1- Discrete Mathematics

Weekly Work Load (in Hrs)	Lecture	Tutorial	Practical
	03	-	-

Online/ In-sem	Theory	Practical	Oral	Term-work	Total Marks	Credit
30	70	-	-	-	100	3

Course Objectives:

To introduce several Discrete Mathematical Structures found to be serving as tools even today in the development of theoretical computer science.

- To introduce students to understand, explain, and apply the foundational mathematical concepts at the core of computer science.
- To understand use of set, function and relation models to understand practical examples, and interpret the associated operations and terminologies in context.
- To acquire knowledge of logic and proof techniques to expand mathematical maturity.
- To learn the fundamental counting principle, permutations, and combinations.
- To study how to model problems using graphs and trees.
- To learn how abstract algebra is used in coding theory.

Course Outcomes:

C201.1 Formulate problems precisely, solve the problems, apply formal proof techniques, and explain the reasoning clearly.

C201.2 Apply appropriate mathematical concepts and skills to solve problems in both familiar and unfamiliar situations including those in real-life contexts.

C201.3 Design and analyze real world engineering problems by applying set theory, propositional logic and to construct proofs using mathematical induction.

C201.4 Specify, manipulate and apply equivalence relations; construct and use functions and apply these concepts to solve new problems.

C201.5 Calculate numbers of possible outcomes using permutations and combinations; to model and analyze computational processes using combinatorics.

C201.6 Model and solve computing problem using tree and graph and solve problems using appropriate algorithms.

C201.7 Analyze the properties of binary operations, apply abstract algebra in coding theory and evaluate the algebraic structures.

Syllabus

UNIT – I

Topics –

Introduction and significance of Discrete Mathematics, **Sets**– Naïve Set Theory (Cantorian Set Theory), Axiomatic Set Theory, Set Operations, Cardinality of set, Principle of inclusion and exclusion. **Types of Sets** – Bounded and Unbounded Sets, Diagonalization Argument, Countable and Uncountable Sets, Finite and Infinite Sets, Countably Infinite and Uncountably Infinite Sets, Power set, **Propositional Logic**- logic, Propositional Equivalences, Application of Propositional Logic- Translating English Sentences, Proof by Mathematical Induction and Strong Mathematical Induction.

Case Study: Know about the great philosophers- Georg Cantor, Richard Dedekind and Aristotle.

UNIT – II

Topics –

Relations and their Properties, n-ary relations and their applications, Representing relations, Closures of relations, Equivalence relations, Partial orderings, Partitions, Hasse diagram, Lattices, Chains and Anti-Chains, Transitive closure and Warshall‘s algorithm. **Functions-**Surjective, Injective and Bijective functions, Identity function, Partial function, Invertible function, Constant function, Inverse functions and Compositions of functions, The Pigeonhole Principle.

Case Study: Know about the great philosophers-Dirichlet

UNIT – III

Topics –

The Basics of Counting, rule of Sum and Product, Permutations and Combinations, Binomial Coefficients and Identities, Generalized Permutations and Combinations, Algorithms for generating Permutations and Combinations.

Case Study: Study Sudoku solving algorithms and algorithm for generation of new SUDOKU. Study Hank-shake Puzzle and algorithm to solve it.

UNIT – IV

Topics –

Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Single source shortest path- Dijkstra's Algorithm, Planar Graphs, Graph Coloring.

Case Study: three utility problems, Web Graph, Google map.

UNIT – V

Topics –

Introduction, properties of trees, Binary search tree, decision tree, prefix codes and Huffman coding, cut sets, Spanning Trees and Minimum Spanning Tree, Kruskal‘s and Prim‘s algorithms, The Max flow- Min Cut Theorem (Transport network).

Case Study: Algebraic Expression Tree, Tic-Tac-Toe Game Tree.

UNIT – VI

Topics –

The structure of algebra, Algebraic Systems, Semi Groups, Monoids, Groups, Homomorphism and Normal Subgroups, and congruence relations, Rings, Integral Domains and Fields, coding theory, Polynomial Rings and polynomial Codes, Brief introduction to Galois Theory –Field Theory and Group Theory.

Text Books

Sr.No		Text Books
1	T1	Kenneth H. Rosen,—Discrete Mathematics and its Applications‖, Tata McGraw-Hill, ISBN 978-0-07-288008-3, 7th Edition.
2	T2	C. L. Liu, —Elements of Discrete Mathematics‖, TMH, ISBN 10:0-07-066913-9.

Reference Books

Sr.No		Text Books
1	R1	Bernard Kolman, Robert C. Busby and Sharon Ross, —Discrete Mathematical Structures, Prentice-Hall of India /Pearson, ISBN: 0132078457, 9780132078450.
2	R2	N. Biggs, — Discrete Mathematics‖, 3rd Edition, Oxford University Press, ISBN 0 –19 850717

Teaching plan

Sr. No.	Unit	Broad Topic to be covered	Books Referred	Total Lectures Planned
1	I	Set Theory and Logic	T1, R1	8
2	II	Relations and Functions,	T1, R1	8
3	III	Counting	T1, R1	8
4	IV	Graph Theory	T1, R1	8
5	V	Trees	T1, T2	8
6	VI	Algebraic Structures and Coding Theory	T1, T2	8

Assessment Tools Details

Sr. No.	Assessment Tool	Total in number	Marks
1	MCQ ESE	01(50 marks)	50
2	MCQ Tests (M1 toM2)	02 (M1-15, M2-15)	30
3	Theory Tests (TH1 to TH2)	02 (each of 15 marks)	30
4	Assignment A1	01 (15 marks)	15
		Total	**105**

Question Bank

Unit-I

1. Explain the principle of duality with example
2. Explain the concept of countably infinite set with example ?
3. It was found that in the first year computer science class of students, 50 knew COBOL, 55 knew C and 46 knew PASCAL. It was also known that 37 knew C and COBOL, 28 knew C and PASCAL and 25 knew PASCAL and COBOL. 7 students however knew none of the languages. Find
    a. How many knew all the three languages
    b. How many knew exactly two languages
    c. how many knew exactly one languages
4. At Sunnydale High School there are 28 students in algebra class, 30 students in biology class, and 8 students in both classes. How many students are in either algebra or biology class?
5. Among the integers 1-1000 :
    a) How many of them are not divisible by 3 nor by 5 nor by 7.
    b) How many of them are not divisible by 5 and 7 but divisible by 3.
6. How many integers between 1-1000 are divisible by 2, 3, 5, or 7 ?
7. Define Tautology and Contradiction.
8. Use mathematical induction to prove that

$1 + 2 + 3 + ... + n = n (n + 1) / 2$,for all positive integers n.

9. Prove that

$1^2 + 2^2 + 3^2 + ... + n^2 = n (n + 1) (2n + 1)/ 6$, For all positive integers n .

10. Use mathematical induction to prove that

$1^3 + 2^3 + 3^3 + ... + n^3 = n^2 (n + 1)^2 / 4$, for all positive integers n.

11. Prove that for any positive integer number n , **$n^3 + 2n$** is divisible by 3
12. Express following statement in propositional form
    a) There are many clouds in the sky but it did not rain
    b) I will get first class if and only if I study well and score above 80 in mathematics
    c) Computers are cheap but softwares are costly
    4. It is very hot and humid or Ramesh is having heart problem
    5. In small restaurants the food is good and service is poor
    6. If I finish my submission before 5.00 in the evening and it is not very hot I will go and play a game of hockey.
    13. Prove that [(p->q) ^ (r->s) ^ (p v r)] -> (q v s) is a tautology
    14. Prove by truth table p -> (Q v R) ≡ (P -> Q) v (P -> R)
    15. show by induction that n>= 1

$1^2 + 3^2 + 5^2 + ... (2n-1)^2 = n(2n-1)(2n+1) / 3$

Unit-II

1. Explain pigeon hole principle?
2. Let f(x) = x + 2, g(x) = x-2, h(x) = 3x find

i)gof ii)fog iii)fof iv)hog v)gog vi)foh vii)hof

3. Show that if seven numbers from 1 to 12 are chosen, then two of them will add up to 13.
4. Prove that among 100,000 people there are bor n at exactly the same time (hour, min. sec.
5. What is function? Explain following functions with diagram

i)Composite function
ii) Injective function
iii)Surjective
iv) Bijective function

6. Let n be a positive integer ,Sn be the set of all divisors of n. let D denote relation of Division . Draw the diagrams of lattices for :

i)n = 6 ii) n = 24 iii) n = 30

7. Determine the following relations are equivalence relation for given matrix

1. [111,011,111]
2. [1010, 0101, 1010, 0101]
3. [1110, 1110, 1110, 0001]

8. Determine whether the relations represented by the following zero-one matrices are partial orders.
   i. [101,110,001]
   ii. [100,010,101]
   iii. [1010, 0110, 0011, 1101]
9. For each of these relations on the set{1,2,3,4}, decide whether it is reflexive, whether it is symmetric, whether it is antisymmetric, and whether it is transitive.
   a) {(2,2),(2,3),(2,4),(3,2),(3,3),(3,4)}
   b) {(1,1),(1,2),(2,1),(2,2),(3,3),(4,4)}
   c) {(2,4),(4,2)}
   d) {(1,2),(2,3),(3,4)}
   e) {(1,1),(2,2),(3,3),(4,4)}
   f) {(1,3),(1,4),(2,3),(2,4),(3,1),(3,4)}
10. Define transitive closure of a relation and find transitive closure of R by Warshall's algorithm where A={1,2,3,4,5,6}and R={(x,y) | |x-y|=2} . Draw necessary diagraphs.
11. Define properties of binary relation.
    For each of following relations on set A= {1, 2, 3, 4} decide whether it is reflexive,

    symmetric, transitive, antisymmetric.

    R1={(1,1),(2,2),(3,3),(4,4)}.

    R2={(1,1),(1,2),(2,1),(2,2),(3,3),(3,4)}.

    R3={(1,3),(1,4),(2,3),(2,4),(3,1),(3,4)}.

12. Define equivalence relation and partial ordering relation. Let A= {1, 2, 3, 4, 5, 6}

    and π = {{1, 2}, {3, 4, 5}, {6}}.

    Find equivalence relation determined by π and draw its diagraph?

13. Let A is the set of all factors of positive integer m and relation is divisibility on A i.e.

    R={(x,y) | x,y $\in$ A, x divides y}.

    For m=45; Fnd whether R is partial ordering relation and Draw Hasse diagram for it.

    If so find whether POSET is lattice? Also find minimal and maximal elements, chains and antichains.

14. For each of these relations on the set{1,2,3,4}, decide whether it is reflexive, whether it is symmetric, whether it is antisymmetric, and whether it is transitive.

a) {(2,2),(2,3),(2,4),(3,2),(3,3),(3,4)}
b) {(1,1),(1,2),(2,1),(2,2),(3,3),(4,4)}
c) {(2,4),(4,2)}
d) {(1,2),(2,3),(3,4)}
e) {(1,1),(2,2),(3,3),(4,4)}
f) {(1,3),(1,4),(2,3),(2,4),(3,1),(3,4)}

15. Find the inverse function of f(x)= x3 +1.

Unit -III

1. How many permutations of the letters ABCDEFG contain
   a) the string BCD
   b) the string CFGA
   c) the strings BA and GF
   d) the strings ABC and DE
   e) the strings ABC and CDE
   f) the strings CBA and BED

2. A committee of 5 people is to be formed from group of 4 men and 7 women.

   How many possible committees can be formed if at least 3 women are on the committee?

3.. How many permutations can be made out of the letter of the word "COMPUTER" ? How many of these

   i) begin with C?
   ii) end with R?
   iii) begin with C and end with R?
   iv) C and R occupy end places?

4. How many seven digit numbers can be formed using digits 1,7,2,7,6,7,6?

5. . How many 4 digit numbers can be formed by using the digits 2,4,6,8 when repetition of digits are Allowed?

6. . How many 4 digit numbers can be formed by using the digits 1,3,4,6,8 when repetition of digits are Allowed?

7. In how many ways 10 programmers can sit on a round table to discuss the project so that project leader and particular programmer always sit together ?

8. In how many ways can 30 late admitted students be assigned to three practical batches A, B, C if A can accommodate 10 students, B – 15 students, C- 5 students only?
9. From 10 programmers in how many ways can 5 be selected when
   i) particular programmer is included every time?
   ii) particular programmer is not included at all?
10. Generate all permutations for n = 3 by next permutation method?
11. Explain basic counting principle with the help of Sum rule and Product rule?

12. How many four digit numbers can be formed out of digits 1,2,3……9, if
    i)No repetition is permitted

    ii)How many of these will be greater than 3000

13. Suppose repetitions are not permittable, how many four digit numbers can be formed From six digits 1,2,3,5,7,8?
    ii)How many of such numbers are less than 4000?
    iii)How many in (i) are even?

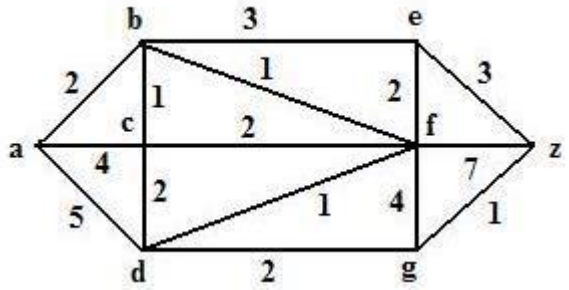
    iv)How many in (i) are odd?

    v)How many in (i) are divisible by 10?

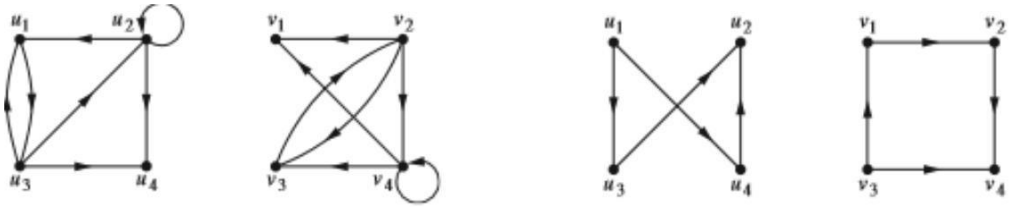
14. How many ways are there to select five players from a 10-member tennis team to make a trip to a match at another school?
15. A group of 30 people have been trained as astronauts to go on the first mission to Mars. How many ways are there to select a crew of six people to go on this mission (assuming that all crew members have the same job)?

Unit-IV

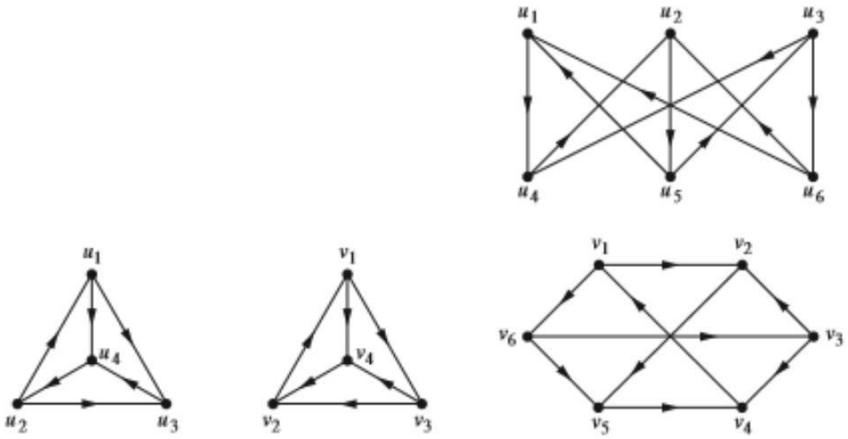
1. With reference to graph theory define the following terms with example.
   a) Eulerianpath and circuit
   b) Planer Graph
   c) Bipartite Graph
   d) Isomorphic graph
2. Find the shortest length of path from a to z in the graph given below by using Dijkstra's Shortest path algorithm.



3. Determine the given graph are isomorphic or not

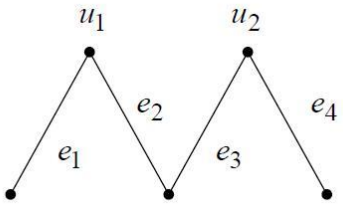


(a) (b)

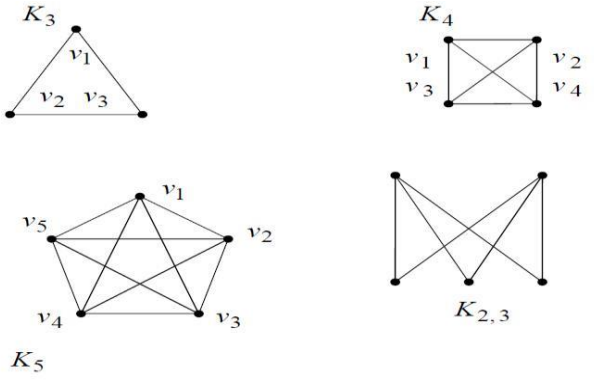


(c) (d)

4. List and explain necessary and sufficient conditions for Euler's path & circuit.
5. Is following graph is bipartite graph? If yes explain.

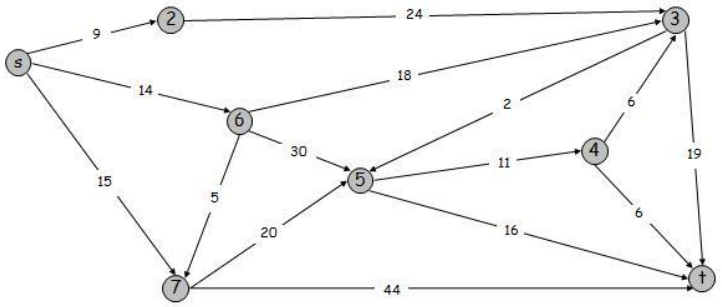


6. Which of these graphs areEulerian?

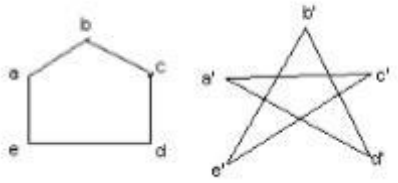


7. Find the shortest length of path from s to t in the graph given below by using Dijkstra's

Shortest path algorithm



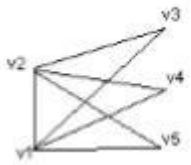
8. What do you mean by graph isomorphism, show it by example?
9. Show that the given graph are isomorphic?

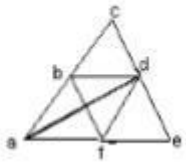


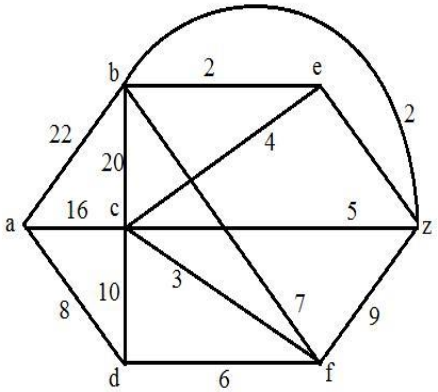
10. Determine whether the following graph contain Eulerian circuit. If it does, then find an Eulerian circuit.

11. Find the shortest length of path from a to z in the graph given below by using Dijkstra's

Shortest path algorithm.



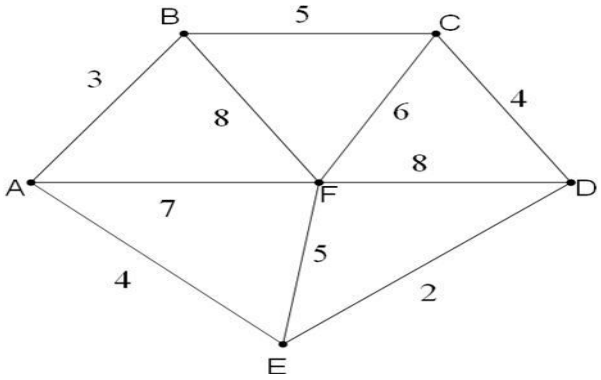




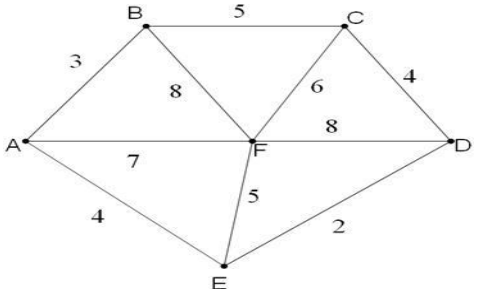
Unit-V

1. Define following terms with suitable example:
   i. prefix code
   ii. minimum spanning tree

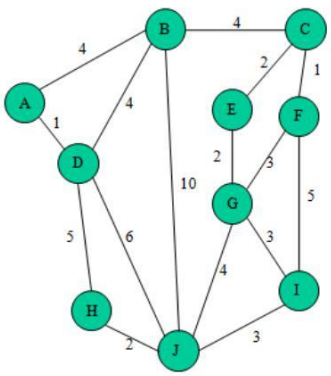
2. Find the minimum spanning tree for the graph given in the following figure using Prim's algorithm.



3. Find the minimum spanning spanning tree for the graph given in the following figure using Kruskal's algorithm.

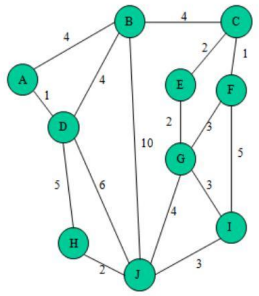


4. Find the minimum spanning spanning tree for the graph given in the following figure using Prim's algorithm.

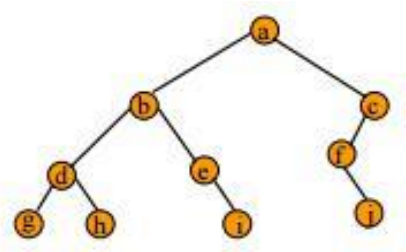


5. Find the minimum spanning spanning tree for the graph given in the following figure using

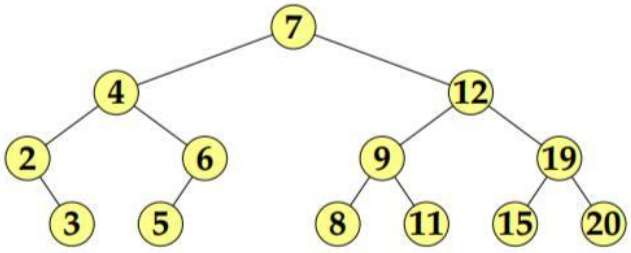
   Kruskal's algorithm.



6. Find the preorder, postorder and inorder traversal of the following tree.



7. Find the preorder, postorder and inorder traversal of the following tree.



8. Define following terms with suitable example:
   i. Eccentricity of vertex
   ii. Binary tree
   iii. Binary tree traversal
9. Construct the labelled tree of the following algebraic expression (((x+y)*z)/3) + (19+(x*x))
10. Represent the expression ((a+5*4)* (b-(5+9)) using a binary tree.
11. For the following set of weights, construct optimal binary tree prefix code. For each weigh in the set, give the corresponding code words. 8,9,12,14,16,19
12. A secondary storage media contains information in files with different formats. The frequency of different types of files is as follows.

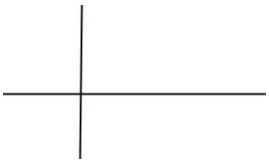
    Exe(20), bin(75), bat(20), jpeg(85), dat(51), doc(32), sys(26), c(19), cpp(25), bmp(30), avi(24), prj(29), 1st(35), zip(37)

Unit-VI

1. Show that the algebraic system (A, +) is a monoid, where A is the set of integers and '+' is a binary operating giving addition of two integers.
2. If set $Q_1$of all rational numbers other than 1 with a*b= a+b-ab. Show that (G, *) is a group.
3. Let G be the set of all non-zero real numbers and let a*b= ab/2. Show that (G, *) is an abelian group.
4. Show that the set G= {1, w, w^2} where w is the cube root of unity is a group with respect to multiplication.
5. Let (A, *) be a group. Show that (A, *) is abelian group.
6. Show that the set of all id impotent in a commutative monoid S is a submonoid of S.
7. Prove that the set Z of all integers with binary operation * defined by a* b=a+b+1 such that □ a,b € Z is an abelian group.
8. Let Z ={0,1,2,3,4,…(n-1)}and '♦' be a binary opeation such that a ♦ b =remainder of abelian when divided by n. Construct a table for n=4, Is (z_4, ♦) is groupoid, monoid, semigroup and abelian group?
9. Prove that every cyclic group is an abelian group.
10. Define following terms:
    i. Homomorphism of groups

    ii. Isomorphism of groups
11. Explain homomorphism and isomorphism of groups with examples.
12. Let G be a group with identity e show that a function f: G---> G defined by f(a) □ = a, € G is a homomorphism (Endomorphism).
13. Define:
    i. Rings
    ii. Domain

    iii. Field
14. Show that S= {a+b$\sqrt{2}$ $a, b \in z$ } for the operations +, * is an integral domain but not a field.
15. Let Z={0,1,2,3,4,5,6,7}. Let R is a relation under the operations addition modulo 7 and multiplication modulo 7. Does this system form a ring? It is a commutative ring?
16. Find the minimum distance of an encoding function e: B^2□B^5 given as:
    e(0 0)= 00000, e(0 1)= 10011, e(1 0)=01110, e(1 1)= 11111.
17. Let G= {even, odd} and binary operation ® is defined as :
    ®even odd



even even odd

odd odd even

Find whether (G,®) is an Abelian group?

18. With reference to algebraic system define the following terms with suitable example.
    a) SubgroupGroup
    d) Homomorphism
    e) Ring

13. Let A= {a,b}. Which of the following tables define a Semigroup on A? Which of define a Monoid on A?

*	a	b
a	a	b
b	a	a

Table I

*	a	b
a	a	b
b	b	b

Table II

20. Define following terms:
    i. Monoid
    ii. Semi group
    iii. Groupoid

2. Course Name – Fundamentals of Data Structure (210242)

Weekly Work Load (in Hrs)	Lecture	Tutorial	Practical
	03	-	04

Online/ In-sem	Theory	Practical	Oral	Term-work	Total Marks	Credit
30	70	50	-	25	175	05

Course Objectives

- To understand the standard and abstract data representation methods.
- To acquaint with the structural constraints and advantages in usage of the data.
- To understand various data structures, operations on it and the memory requirements
- To understand various data searching and sorting methods.
- To understand various algorithmic strategies to approach the problem solution.

Course Outcomes: At the end of the course, students will be able to,

C202.1 Design the algorithms to solve the programming problems, identify appropriate algorithmic strategy for specific application, and analyze the time and space complexity.

C202.2 Discriminate the usage of various structures, Design/Program/Implement the appropriate data structures; use them in implementations of abstract data types and Identity the appropriate data structure in approaching the problem solution.

C202.3 Demonstrate use of sequential data structures- Array and Linked lists to store and process data.

C202.4 Understand the computational efficiency of the principal algorithms for searching and sorting and choose the most efficient one for the application.

C202.5 Compare and contrast different implementations of data structures (dynamic and static).

C202.6 Understand, Implement and apply principles of data structures-stack and queue to solve computational problems.

Syllabus

Unit	Course Contents	Hours
I	**Introduction to data structures and algorithms**	07
	Introduction: From Problem to Program (Problem, Solution, Algorithm, Data Structure and Program). Data Structures: Data, Information, Knowledge, and Data structure, Abstract Data Types (ADT), Data Structure Classification (Linear and Non-linear, Static and Dynamic, Persistent and Ephemeral data structures). **Algorithms:** Problem Solving, Introduction to algorithm, Characteristics of algorithm, Algorithm design tools: Pseudo-code and flowchart. **Complexity of algorithm:** Space complexity, Time complexity, Asymptotic notation- Big-O, Theta and Omega, finding complexity using step count method, Analysis of programming constructs-Linear, Quadratic, Cubic, Logarithmic. **Algorithmic Strategies:** Introduction to algorithm design strategies- Divide and Conquer, and Greedy strategy.	
II	**Linear Data Structures and using Sequential organization**	07
	Concept of Sequential Organization, Overview of Array, Array as an Abstract Data Type, Operations on Array, Merging of two arrays, Storage Representation and their Address Calculation: Row major and Column Major, Multidimensional Arrays: Two-dimensional arrays, n-dimensional arrays. Concept of Ordered List, **Single Variable Polynomial**: Representation using arrays, Polynomial as array of structure, Polynomial addition, Polynomial multiplication.	
III	**Searching and Sorting**	07
	Searching: Search Techniques-Sequential Search/Linear Search, Variant of Sequential Search- Sentinel Search, Binary Search, Fibonacci Search, and Indexed Sequential Search. Sorting: Types of Sorting-Internal and External Sorting, General Sort Concepts-Sort Order, Stability, Efficiency, and Number of Passes, Comparison Based Sorting Methods-Bubble Sort, Insertion Sort, Selection Sort, Quick Sort, Shell Sort, Non-comparison Based Sorting Methods-Radix Sort, Counting Sort, and Bucket Sort, Comparison of All Sorting Methods and their complexities	
IV	**Linked List**	07
	Introduction, of Linked Lists, Realization of linked list using dynamic memory management, operations, Linked List as ADT, **Types of Linked List:** singly linked, linear and Circular Linked Lists, Doubly Linked List, Doubly Circular Linked List, Primitive Operations on Linked List- Create, Traverse, Search, Insert, Delete, Sort, Concatenate. Polynomial Manipulations-Polynomial addition. Generalized Linked List (GLL) concept, Representation of Polynomial using GLL.	

V	**Stack**	**07**
	Basic concept, stack Abstract Data Type, Representation of Stacks Using Sequential Organization, stack operations, Multiple Stacks, Applications of Stack- Expression Evaluation and Conversion, Polish notation and expression conversion, Need for prefix and postfix expressions, Postfix expression evaluation, Linked Stack and Operations. Recursion- concept, variants of recursion- direct, indirect, tail and tree, backtracking algorithmic Strategy, use of stack in backtracking.	
VI	**Queue**	**07**
	Basic concept, Queue as Abstract Data Type, Representation of Queue using Sequential organization, Queue Operations, Circular Queue and its advantages, Multi-queues, Linked Queue and Operations. Deque-Basic concept, types (Input restricted and Output restricted), Priority Queue-Basic concept, types (Ascending and Descending).	

Text Books

1. Horowitz, Sahani, Dinesh Mehata, ―Fundamentals of Data Structures in C++.
2. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, "Data Structures and Algorithms in Python".

Reference Books

1. Sartaj Sahani, ―Data Structures, Algorithms and Applications in C++‖,.
2. G A V Pai, ―Data Structures and Algorithms
3. Allen Downey, Jeffery Elkner, Chris Meyers, "How to think like a Computer Scientist: Learning with Python

2.5 Teaching plan

Sr. No.	Unit	Broad Topic to be covered	Total Lectures Planned
1	Introduction to Algorithm and Data Structures	Basics of Algorithms and time complexity	7
2	Linear Data Structures Using Sequential Organization	Basics of Linear data structures and introduction to implementation of linear data structures	7
3	Sorting and Searching	Types of searching and sorting methods	7
4	Linked Lists	Basics of linked list and its operations	7

5	Stacks	Basics of stacks and implantation ,basic operations and introductions to stacks applications	7
6	Queues	Basics of stacks and implantation ,basic operations and introductions to stacks applications	7

Assessment Tools Details

Sr. No.	Assessment Tool	Total in number	Marks
1	MCQ ESE	01(50 marks)	50
2	MCQ Tests (M1 toM2)	02 (M1-15, M2-15)	30
3	Theory Tests (TH1 to TH2)	02 (each of 15 marks)	30
4	Assignment A1	01 (15 marks)	15
		Total	**105**

Assessment Tool Planner

Class – SE
Course Name – Fundamentals of Data Structures
Course Code – 210242 **Course No.** – 202
Teaching Scheme **Marking Scheme**
Theory – 3Hrs/wk Theory Marks
MSE – 30**ESE** – 70 ----

Detail Schedule / Plan of conduction of assessment tool:

Units	Assessment Tool	Marks	Schedule
I	Assignment 1	10	1st week of July 2021
I & II	MCQ Test 1	20	2nd week of July 2021
III & IV	MCQ Test 2	20	1st week of Aug 2021
V&VI	Test3	20	2nd week of sep 2021

Unit wise Question Bank

Unit-I

- Define and explain the following terms : (i) Data (ii) Data structure (iii) Flowchart.
- Define algorithms and explain its characteristics.
- Explain the divide and conquer strategy with suitable example. Comment on its time complexity.
- Explain the Asymptotic notation Big O, Omega and Theta with suitable example.
- Explain static and dynamic data structures with examples.
- Differentiate between linear and non-linear data structure with example.
- Explain the Greedy strategy with suitable example. Comment on its time complexity

Unit II

- Define & explain the term i) Sequential Organization. ii) Ordered List
- Explain polynomial representation using arrays with suitable example
- Derive address calculation formula for one-dimensional array with one example.
- Explain two-dimensional arrays with row and column major implementation. Explain address calculation in both cases with example.
- What is sparse matrix ? Explain with suitable example.
- Explain fast Transpose of sparse matrix with suitable example, Discuss time complexity of fast transpose.
- Write pseudo Python code to perform simple transpose of sparse matrix. Discuss its time complexity.
- Write pseudo Python code to perform polynomial multiplication using array.
- Write pseudo Python code for reversing a string and state its time complexity.

.

UNIT III

- Explain Linear search and binary search with example. State its time complexity and compare linear and binary search(time and space complexity).
- Write an algorithm for searching an element using binary search. Discuss the time complexity of algorithm in best case and worst case.
- Write an algorithm for Fibonacci search and find out its time complexity.
- Explain merge sort algorithm using divide and conquer strategy with an example. State its time complexity and space complexity.
- Explain the algorithm of Quick sort with suitable example. Discuss its time complexity and space complexity.
- Write short note on stability of sorting. Compare bubble, insertion and selection sort with one example and discuss time complexity.
- What is heap ? Explain heap sort with suitable example. State its complexity.
- Compare Heap sort and Quick sort with one example and discuss time complexity.

- Explain insertion sort algorithm and sort the given list using insertion sort : 1) List : 7, 4, 10, 6, 3, 12, 1, 8, 2, 15, 9, 5 2) List : 55, 85, 45, 11, 34, 05, 89, 99, 67 Discuss its time complexity and space complexity.
- 10.Explain quick sort and Sort the following numbers using quick sort : State its time complexity and space complexity.

  1) List : 39, 09, 81, 45, 90, 27, 72, 18

  2) List : 25, 82, 17, 23, 38, 7, 64, 86, 21

  3) List : 15, 08, 20, -4, 16, 02, 01, 12, 21, -2

UNIT IV

- Compare Singly, Doubly and Circular Linked List.
- Explain Generalized Linked List with suitable example.
- Explain polynomial representation using linked list with an example.
  .Represent the following using GLL : (p, q(r, s(u, v), w) (x, y))
- Represent the following polynomial by using-generalized linked list : (a, b (c, d (e, g,), h) (f))
- Write a pseudo C++ code to reverse singly linked list
- Write pseudo C++ code to represent singly linked list as an ADT.
  Write pseudo C++ code to represent doubly linked list as an ADT.
- Write pseudo C++ code to represent circular linked list as an ADT.
- Write pseudo C++ code for polynomial addition using singly linked list.

UNIT V

- What is stack ? Write an ADT for stack.
- What is recursion ? Explain use of stack for recursion.
- Explain (algo.) evaluation of postfix expression using stack with example.
- Write algorithm to convert infix expression to postfix expression.
- Define Backtracking & Explain use of backtracking in 4-Queen's problem.
- Give pseudo C++ code to implement the foll. operations on linked stack : (i) Create (ii) Push data
- Explain the stepwise conversion using stack for the given infix expression to the postfix expression : A * B + C * D.
- Explain the stepwise conversion using stack for the given infix expression to the postfix expression : A * (B + C) * D.
- Explain process of conversion of an infix expression to postfix expression using stack : A * (B – C)/E ^ F + G.
- 10.Explain the stepwise conversion using stack for the given infix expression to the postfix expression : ((a/(b-c+d))*(e-a)*c

UNIT VI

- Write pseudo C++ code to implement a simple queue using Array, OR Write pseudo C++ code to perform insert and delete operation on linear queue.
- Write pseudo C++ code to implement Queue using Linked list.
- Explain Dequeue with the insert and delete operations performed on it.
- What is Priority queue ? Describe the operations on priority queue and explain its applications. OR Explain priority queue. Write ADT for priority queue and state its applications
- Write pseudo C++ code to implement circular queue using array.
- Explain circular queue using linked list. Write pseudo C code for enqueue operation.
- Explain linear queue and circular queue with suitable example. Give the advantages of circular queue over linear queue.
- Explain priority queue. Give pseudo C++ code for array implementation of priority queue.
- Explain the application of queue in details.
- 10 .What is circular queue ? Write a c function to insert element into the circular queue.

3. Name of Course: Object Oriented Programming (210243)

Weekly Work Load (in Hrs)	Lecture	Tutorial	Practical
	03	-	04

Online/ In-sem	Theory	Practical	Oral	Term-work	Total Marks	Credit
30	70	50	-	25	175	05

Course Objectives

- To explore the principles of Object Oriented Programming (OOP).
- To understand object-oriented concepts such as data abstraction, encapsulation, inheritance, dynamic binding, and polymorphism.
- To use the object-oriented paradigm in program design.
- To lay a foundation for advanced programming.
- Provide programming insight using OOP constructs.

Course Outcomes

At the end of the Course, Students will be able to,

C203.1 Apply constructs- sequence, selection and iteration; classes and objects, inheritance, use of predefined classes from libraries while developing software.

C203.2 Design object-oriented solutions for small systems involving multiple objects.

C203.3 Use virtual and pure virtual function and complex programming situations.

C203.4 Apply object-oriented software principles in problem solving.

C203.5 Analyze the strengths of object-oriented programming.

C203.6 Develop the application using object oriented programming language (C++).

Syllabus

Unit	Course Contents	Hours
I	**Introduction to object-oriented programming**	**07**
	Introduction to object-oriented programming, Need of object-oriented programming, Fundamentals of object-oriented programming: Namespaces, objects, classes, data members, methods, messages, data encapsulation, data abstraction and information hiding, inheritance, polymorphism. Benefits of OOP, C++ as object oriented programming language. **C++ Programming**- C++ programming Basics, Data Types, Structures, Enumerations, control structures, Arrays and Strings, Class, Object, class and data abstraction, Access specifiers, separating interface from implementation. **Functions**- Function, function prototype, accessing function and utility function, Constructors and destructor, Types of constructor, Objects and Memory requirements, Static members: variable and functions, inline function, friend function.	
II	**Inheritance & Pointers**	**07**
	Inheritance- Base Class and derived Class, protected members, relationship between base Class and derived Class, Constructor and destructor in Derived Class, Overriding Member Functions, Class Hierarchies, Public and Private Inheritance, Types of Inheritance, Ambiguity in Multiple Inheritance, Virtual Base Class, Abstract class, Friend Class, Nested Class. **Pointers:** declaring and initializing pointers, indirection Operators, Memory Management: new and delete, Pointers to Objects, this pointer, Pointers Vs. Arrays, accessing Arrays using pointers, Arrays of Pointers, Function pointers, Pointers to Pointers, Pointers to Derived classes, Passing pointers to functions, Return pointers from functions, Null pointer, void pointer.	
III	**Polymorphism**	**07**
	Polymorphism- Introduction to Polymorphism, Types of Polymorphism, Operator Overloading-concept of overloading, operator overloading, Overloading Unary Operators, Overloading Binary Operators, Data Conversion, Type casting (implicit and explicit), Pitfalls of Operator Overloading and Conversion, Keywords explicit and mutable. Function overloading, **Run Time Polymorphism**- Pointers to Base class, virtual function and its significance in C++, pure virtual function and virtual table, virtual destructor, abstract base class.	
IV	**Files**	**07**

	Data hierarchy, Stream and files, Stream Classes, Stream Errors, Disk File I/O with Streams, File Pointers, and Error Handling in File I/O, File I/O with Member Functions, Overloading the Extraction and Insertion Operators, memory as a Stream Object, Command-Line Arguments, Printer output.	
V	**Exception Handling & Template**	**07**
	Exception Handling- Fundamentals, other error handling techniques, simple exception handling- Divide by Zero, Multiple catching, re-throwing an exception, exception specifications, user defined exceptions, processing unexpected exceptions, constructor, destructor and exception handling, exception and inheritance. **Templates**- The Power of Templates, Function template, overloading Function templates, and class template, class template and Nontype parameters, template and friends Generic Functions, The type name and export keywords.	
VI	**STL**	**07**
	Introduction to STL, STL Components, Containers- Sequence container and associative containers,container adapters, Application of Container classes: vector, list **Algorithms**- basic searching and sorting algorithms, min-max algorithm, set operations, heap sort, **Iterators**- input, output, forward, bidirectional and random access. Object Oriented Programming – a road map to future	

Text Books

1. Deitel,"C++HowtoProgram",4thEdition,PearsonEducation,ISBN:81-297-0276-2
2. RobertLafore,"ObjectOrientedProgramminginC++‖",fourthedition,SamsPublishing, ISBN:0672323087 (ISBN 13:9780672323089

Reference Books

1. Herbert Schildt, "C++-The complete reference"‖, Eighth Edition, McGraw Hill Professional, 2011,ISBN:978-00-72226805
2. Matt Weisfeld, "The Object-Oriented Thought Process", Third Edition PearsonISBN-13:075- 2063330166
3. E.Balagurusamy, "Object-Oriented Programming with C++", 7th edition, Graw-Hill Publication, ISBN 10: 9352607996 ISBN 13: 9789352607990
4. Cox Brad, Andrew J. Novobilski, "Object–Oriented Programming: An EvolutionaryApproach"‖, Second Edition, Addison–Wesley, ISBN:13:978-020-1548341

Teaching plan

Sr. No.	Unit	Broad Topic to be covered	Total Lectures Planned
1	Introduction to C++ Programming	C++ Programming, Functions	07
2	Knowledge Delivery	Inheritance and Pointers	07
3	Polymorphism	Polymorphism- Run Time Polymorphism	07
4	Files	Files and Streams-Modes of operation of files	07
5	Exception Handling	Exception Handling and Templates-Divide by Zero, Multiple catching	07
6	STL	Standard Template Library (STL)-Components, Containers, Algorithms	07

Assignment Tool Details

Sr. No.	Assessment Tool	Total in number	Marks
1	Unit Test 1	01(50 marks)	15
2	MCQ Test1	02 (M1-15, M2-15)	15
3	Assignment 1	02 (each of 15 marks)	20
4	Unit Test 2	01 (15 Marks)	15
5	MCQ Test2	01 (20 Marks)	20
6	Skill set	01 (15 marks)	15
		Total	**100**

Assessment Tool Planner

Class – SE
Course Name – Object Oriented Programming
Course Code – (210243) **Course No.** – 203
Teaching Scheme **Marking Scheme**
Theory – 3Hrs/wk Theory Marks
MSE – 30**ESE** – 70

Detail Schedule / Plan of conduction of assessment tool:

Units	Assessment Tool	Marks	Schedule
I	Unit Test 1	15	4th week of August 2021
I & II	MCQ Test1	15	4th week of Sep 2021
III	Assignment 1	20	Last week of Oct 2021
III & IV	Unit Test 2	15	1st week of November 2021
All Units	MCQ Test2	20	2nd week of December 2021
All Units	Skill set	15	2nd week of December2021

Practical Assignment

Sr. No.	Title
	Group A
LA1	Implement a class Complex which represents the Complex Number data type. Implement the following . Constructor (including a default constructor which creates the complex number 0+0i). . Overload operator+ to add two complex numbers. . Overload operator* to multiply two complex numbers. Overload operators << and >> to print and read Complex Numbers.
LA2	Develop a program in C++ to create a database of student's information system containing the following information: Name, Roll number, Class, Division, Date of Birth, Blood group, Contact address, Telephone number, Driving license no. and other. Construct the database with suitable member functions. Make use of constructor, default constructor, copy constructor, destructor, static member functions, friend class, this pointer, inline code and dynamic memory allocation operators-new and delete as well as exception handling.
LA3	Imagine a publishing company which does marketing for book and audio cassette versions. Create a class publication that stores the title (a string) and price (type float) of publications. From this class derive two classes: book which adds a page count (type int) and tape which adds a playing time in minutes (type float). Write a program that instantiates the book and tape class, allows user to enter data and displays the data members. If an exception is caught, replace all the data member values with zero values.
	Group B
LA4	Write a C++ program that creates an output file, writes information to it, closes the file, open it again as an input file and read the information from the file.
LA5	Write a function template for selection sort that inputs, sorts and outputs an integer array and a float array.
	Group C
LA6	Write C++ program using STL for sorting and searching user defined records such as personal records (Name, DOB, Telephone number etc) using vector container.
LA7	Write a program in C++ to use map associative container. The keys will be the names of states and the values will be the populations of the states. When the program runs, the user is prompted to type the name of a state. The program then looks in the map, using the state name as an index and returns the population of the state.
	Group B

LA8	Write C++ program to draw 2-D object and perform following basic transformations, Scaling b) Translation c) Rotation. Apply the concept of operator overloading.
	Group C
LA9	Write a C++ program to implement bouncing ball using sine wave form. Apply the concept of polymorphism.

Unit wise Question Bank

Unit 1.

- Describe the following characteristics of OOP
  - i.Encapsulation
  - ii. Polymorphism,
  - iii. Inheritance
- Discuss function prototyping, with an example. Also write its advantage.
- Write the general form of function. Explain the different types of argument passing techniques with example.
- Why C++ introduced reference variable?
- Give the comparison of C and C++ with examples.
- What are pointers explain with an example.
- What is function overloading give example?
- Differentiate between procedures oriented and object oriented programming.
- Write a C++program to count the number of objects of a certain class.
- What are constructors? How are they different from member functions?

Unit II

- Explain different types of inheritance with block diagram and an example for each
- What is the ambiguity that arises in multiple inheritances? How it can be overcome. Explain with example.
- Discuss with examples, the implications of deriving a class from an existing class by the ‘public’and‘protected’ access specifiers.
- Write a c++ program to initialize base class members through a derived class constructor.
- What is inheritance? How to inherit a base class as protected?
- With an example explain, multiple base class inheritance?
- What is the need of virtual function? With an example, explain overriding of Member function of base class in derived class?
- What is the virtual destructor?
- List the library classes that handle streams in c++.

UNIT III

- Explain error handling and manipulators inc++?
- Why friend f unction is required to overload binaryoperators?

- What are the rules for overloadingoperators?
- Write the difference between Early and LateBinding.
- Explain Pure VirtualFunctions.
- Explain Calling a Virtual Function Through a Base ClassReference.
- Define a class Date, use overloaded + operator to add two dates and display the result. Assume non leap yeardates.
- Write a program to demonstrate friend function inC++.
- What is the need of overloading operators andfunctions?
- Write down the example to overload unary and binary operators inC++.

UNIT IV

- Explain C++ Stream Classes.
- Explain Formatted I/O.
- Explain File operations.
- Explain ignore (), flush (), peek () and putback ()functions.
- What are the filestreams?
- Explain the process of open,read,write and close files?
- Explain the role of seekg (), seekp (),tellg(),tellp(),function in the process of random access in a file.
- Explain the Standard Template Library and how it isworking.
- Write a C++ program using function template to find the product of two integer or floating point type of data.

UNIT V

- What are the new style cast operators explain the syntax of these operators with example ?
- What are class templates.? How are they created? What is the need for class templates? Create a template for bubble sort functions.
- Explain the C++style solution for handling exceptions
- Explain try catch and throw exception handling inc++
- Explain different types of type conversion.
- Explain with example, how Function Templates are implemented?
- What are the file strams?
- Explain the process of open, read, write and close files?

- Explain the role of seekg(),seekp(),tellg(),tellp(),function in the process of random access in a file.
- Explain the Standard Template Library and how it is working.

UNIT VI

- What's std::vector and why should I use std::vector?
- 5 Different ways to Initialize a vector
- How to Iterate or Loop over a Vector?
- How to Iterate over a Vector in Reverse Order (Backward direction)
- How does std::vector works internally ?
- How to fill a vector with random numbers in C++
- Importance of Constructors while using User Defined Objects with std::vector
- How to use vector efficiently in C++?
- std::vector and Iterator Invalidation

4. Name of Course: Computer Graphics (210244)

Weekly Work Load (in Hrs)	Lecture	Tutorial	Practical
	03	-	04

Online/ In-sem	Theory	Practical	Oral	Term-work	Total Marks	Credit
30	70	50	-	25	175	05

Course Objectives

- Remembering: To acquaint the learner with the basic concepts of Computer Graphics.
- Understanding: To learn the various algorithms for generating and rendering graphical figures.
- Applying: To get familiar with mathematics behind the graphical transformations.
- Understanding: To understand and apply various methods and techniques regarding projections, animation, shading, illumination and lighting.
- Creating: To generate Interactive graphics using OpenGL.

Course Outcomes:

On completion of the course, learner will be able to,

C204.1 Identify the basic terminologies of computer Graphics and interpret the mathematical foundation of the concepts of computer graphics.

C204.2 Apply mathematics to develop Computer programs for elementary graphic operations.

C204.3 Illustrate the concepts of windowing and clipping and apply various algorithms to fill and clip polygons.

C204.4 Understand and apply the core concepts of computer graphics, including transformation in two and three dimensions, viewing and projection

C204.5 Understand the concepts of color models, lighting, shading models and hidden surface elimination.

C204.6 Create effective programs using concepts of curves, fractals, animation and gaming

Syllabus

Unit	Course Contents	Hours
I	**Graphics Primitives and Scan**	**06**
	Introduction, graphics primitives - pixel, resolution, aspect ratio, frame buffer. Display devices, applications of computer graphics. Introduction to OpenGL - OpenGL architecture, primitives and attributes, simple modeling and rendering of two- and three-dimensional geometric objects, GLUT, interaction, events and call-backs picking. (Simple Interaction with the Mouse and Keyboard) Scan conversion: Line drawing algorithms: Digital Differential Analyzer (DDA), Bresenham. Circle drawing algorithms: DDA, Bresenham, and Midpoint.	
II	**Polygon, Windowing and Clipping**	**07**
	Polygons: Introduction to polygon, types: convex, concave and complex. Inside test. Polygon Filling: flood fill, seed fill, scan line fill. Windowing and clipping: viewing transformations, 2-D clipping: Cohen – Sutherland algorithm line Clipping algorithm, Sutherland Hodgeman Polygon clipping algorithm, Weiler Atherton Polygon Clipping algorithm.	
III	**2D, 3D Transformations and Projections**	**07**
	2-D transformations: introduction, homogeneous coordinates, 2-D transformations - Translation, scaling, rotation and shear, rotation about an arbitrary point.3-D transformation: introduction, 3-D transformations - Translation, scaling, rotation and shear, rotation about an arbitrary axis. Projections : Parallel (Oblique: Cavalier, Cabinet and orthographic: isometric, diametric, trimetric) and Perspective (Vanishing Points – 1 point, 2 point and 3 point)	
IV	**Light, Colour, Shading and Hidden Surfaces**	**06**
	Colour models: Properties of Light, CIE chromaticity Diagram, RGB, HSV, CMY. Illumination Models: Ambient Light, Diffuse reflection, Specular Reflection, and the Phong model, Combined diffuse and Specular reflections with multiple light sources, warn model, Shading Algorithms: Halftone, Gauraud and Phong Shading. Hidden Surfaces Introduction, Back face detection and removal, Algorithms: Depth buffer (z), Depth sorts (Painter), Area subdivision (Warnock)	
V	**Curves and Fractals**	**06**

	Curves: Introduction, Interpolation and Approximation, Blending function, B-Spline curve, Bezier curve, Fractals: Introduction, Classification, Fractal generation: snowflake, Triadic curve, Hilbert curve, Applications.	
VI	**Introduction to Animation and Gaming**	**06**
	Segment: Introduction, Segment table, Segment creation, closing, deleting and renaming, Visibility. Animation: Introduction, Conventional and computer based animation, Design of animation sequences, Animation languages, Key-frame, Morphing, Motion specification. Gaming: Introduction, Gaming platform (NVIDIA, i8060), Advances in Gaming.	

Course Objective

- Remembering: To acquaint the learner with the basic concepts of Computer Graphics
- Understanding: To learn the various algorithms for generating and rendering graphical
- figures.
- Applying: To get familiar with mathematics behind the graphical transformations
- Understanding: To understand and apply various methods and techniques regarding
- projections, animation, shading, illumination and lighting
- Creating: To generate Interactive graphics using OpenGL

Text Books

1. S. Harrington, “Computer Graphics”‖, 2nd Edition, McGraw-Hill Publications, 1987, ISBN 0
2. Donald D. Hearn and Baker, “Computer Graphics with OpenGL”, 4th Edition, ISBN-13: 9780136053583.
3. D. Rogers, “Procedural Elements for Computer Graphics”, 2nd Edition, Tata McGraw-Hill Publication, 2001, ISBN 0 – 07 – 047371 – 4.
4.

Ref Books

1. J. Foley, V. Dam, S. Feiner, J. Hughes, “Computer Graphics Principles and Practice”‖, 2nd Edition, Pearson Education, 2003, ISBN 81 – 7808 – 038 – 9.
2. D. Rogers, J. Adams, “Mathematical Elements for Computer Graphics”‖, 2nd Edition, Tata McGraw Hill Publication, 2002, ISBN 0 – 07 – 048677 – 8.

Teaching plan

Sr. No.	Unit	Broad Topic to be covered	Total Lectures Planned
1	I	Plotting Primitives, Scan Conversion, Display Files	07
2	II	polygon filling algorithms, Windowing and clipping	07
3	III	2-D transformations, 3-D transformations, Projections	07
4	IV	Color models and Illumination Models, Shading Algorithm, Hidden Surface.	07
5	V	Curves, Fractals	07
6	VI	Segment, Animation, Gaming	07

Assignment Tool Details

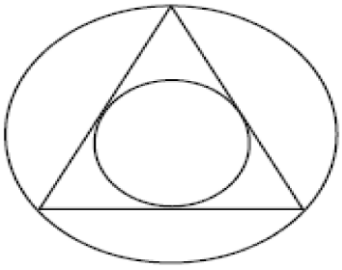
Sr. No.	Assessment Tool	Total in number	Marks
1	MCQ ESE	01(50 marks)	50
2	MCQ Tests (M1 toM2)	02 (M1-15, M2-15)	30
3	Theory Tests (TH1 to TH2)	02 (each of 15 marks)	30
4	Assignment A1	01 (15 marks)	15
		Total	**105**

Assessment Tool Planner

Class – SE
Course Name – Computer Graphics
Course Code – (210244) **Course No.** – 203
Teaching Scheme **Marking Scheme**
Theory – 3Hrs/wk Theory Marks
MSE – 30**ESE** – 70

Sr. No.	Assessment Tool	Marks	Schedule
1	MCQ ESE	50	4th week of August 2021
2	MCQ Tests (M1 toM2)	30	4th week of Oct 2021
3	Theory Tests (TH1 to TH2)	30	Last week of Nov 2021
4	Assignment A1	15	2nd week of December 2021

Practical Assignment:

Sr. No.	Title
	Group A
	PART II Computer Graphics (Group A)
LA1	Write C++ program to draw a concave polygon and fill it with desired color using scan fill algorithm. Apply the concept of inheritance.
LA2	Write C++ program to implement Cohen Southerland line clipping algorithm.
LA3	Write C++ program to draw the following pattern. Use DDA line and Bresenham‘s circle drawing algorithm. Apply the concept of encapsulation 
	Group B
LA4	Write C++ program to draw 2-D object and perform following basic transformations, Scaling b) Translation c) Rotation. Apply the concept of operator overloading.
LA5	Write C++ program to generate Hilbert curve using concept of fractals.
	Group C
LA6	Write OpenGL program to draw Sun Rise and Sunset.

Unit wise Question bank

UNIT-I

1. Define Computer Graphics
2. Define persistence, resolution and aspect ratio.
3. What is horizontal and vertical retrace?
4. What is a raster scan system?
5. What is a random scan system?
6. Write down the attributes of characters.(AU MAY/JUNE 2012 IT)
7. Digitize a line from (10,12) to (15,15) on a raster screen using Bresenham's straight line algorithm.
8. What is antialiasing?
9. What do you mean by emissive and non-emissive displays?
10. What do you mean by scan conversion?
11. What is an output primitive?
12. Explain refresh cathode ray tube.
13. Explain color CRT monitors.
14. Explain direct view storage tubes and liquid crystal displays.
15. Write short notes on Raster scan systems.
16. Describe in detail about the DDA scan conversion algorithm?
17. Write down and explain the midpoint circle drawing algorithm
18. Explain Ellipse generating Algorithm.
19. Explain in detail about Bresenhams line generating algorithm. Give example.
20. Explain in detail about Bresenham's circle generating algorithm. Give example.
21. Explain in detail about Bresenham's ellipse generating algorithm. Give example.
22. Explain in detail about video display devices.
23. Explain in detail about raster and random scan systems.
24. Write a short note on working of raster scan display system and random scan display system.
25. Explain working of Display file interpreter and Video controller.
26. Write a note on working of Cathode Ray Tube.
27. Explain Shadow mask and beam penetration method.
28. What is display file? Explain display file structure.
29. Explain DDA line drawing algorithm with its drawbacks.
30. Compute points on line using mid-point line drawing algorithm for a line passing
31. through end points A(2, 3) and B (10, 8)
32. Explain bresanham's line drawing algorithms.
33. Explain midpoint Circle algorithm.
34. Compute points on arc of circle using midpoint circle drawing algorithm for a circle with radius R = 10 Explain midpoint ellipse algorithm.
35. Digitize the line from (10,16) to (16,12) using the DDA algorithm.
36. Explain DDA line drawing algorithm.

37. Explain Bresenham's line drawing algorithm. How is it advantageous when compared to other existing methods?
38. What do you understand by point? Derive the equation for the intercept form of the line.Find out the conditions for which two lines are perpendicular or parallel.
39. What is a line segment? Also explain the length of the segment. A line segment have end points (2, 1) and (4, 2).Find whether the point(8,4) lies on this line segment or not.
40. Find the distance between point(X0,Y0) and a line rx+sy+t=0 in a plane.
41. What do you understand by plane? Also define the implicit form and explicit form of plane. For the plane 3x+y+5x=0, check the point (1,1,-1)lies in which region R+,R- or on the plane.
42. What is pixels? Explain the frame buffer. In 600*400 pixel, how may K bytes does a frame buffer need?
43. Explain the concept of vector generation of line.
44. Describe DDA line drawing algorithm.
45. Implement the DDA algorithm to draw a line from (0, 0) to (6, 6).
46. Write steps required to draw a line from point(x1, y1) to (x2, y2) using Bresenham's line drawing algorithm. For 10*10 frame buffer, interpret the Bresenham's algorithm to find which pixels are tured on for the line segment (1, 2) and (7, 6).
47. Find out raster location (pixels) which would be chosen by Bresenham's algorithm for a line drawing from screen coordinate (1, 1) to screen coordinate (8, 5).
48. Using Bresenham's line drawing algorithm, draw a line whose end point is (4, 4) and start point is (-3, 0). Q1. What are the different methods of character generation?
49. Differentiate between stoke method and dot-matrix or bitmap method.
50. Explain the Raster Scan System. OR
51. Differentiate between Raster Scan System and Random Scan System.
52. What do you understand by horizontal and vertical retrace in raster system.
53. What is interlacing? Describe the working CRT with suitable diagram and also compare Raster scan CRT with vector scan CRT.
54. Explain beam–penetration and shadow mark method and write down the basic problem with them.
55. What is Computer Graphics? Also explain types and applications of computer graphics?
56. Differentiate between Raster and Vector Graphics? Also explain types of display?
57. How much time is spent in scanning across each row of the pixels during screen on a raster system with a resolution of 640 by 480 and a refresh rate of 60 frames per second?
58. Write all the steps of mid-point circle generating algorithm?
59. Draw a line by using Bresenham's algorithm from (1, 2) to (7, 6)? Brenham's line algorithm is efficient algorithm than DDA why?
60. Write the short note on: DVST Plasma Panel LCD

UNIT-II

1. Distinguish between convex and concave polygons?
2. What is seed fill?
3. Explain and write pseudo code for boundary fill algorithm for polygon filling.
4. Find the normalization transformation window to viewport with window, lower left corner (1,1) and upper right corner at (3,5) onto a viewport for entire normalized device screen.
5. Explain with an example the Cohen-Sutherland line clipping algorithm.
6. Explain Cohen-Sutherland Line clipping with the help of suitable example.
7. What is inside test? Write significance of it. Explain winding number method for inside test.
8. Explain Cohen Sutherland line clipping algorithm.
9. Explain Sutherland Hodgeman polygon clipping algorithm.
10. Explain Boundary fill and Flood fill algorithm.
11. Explain Scan line fill polygon filling algorithm.
12. Explain midpoint subdivision line clipping algorithm with suitable example.
13. Clip a line P1(70,20) and P2(100,40) using Cohen_sutherland algorithm against a window lower left corner (50,10) and upper right corner (80, 40)
14. Suppose there is a rectangle ABCD whose coordinates are A(1,1), B(4,1), C(4,4), D(1,4) and
15. the window coordinates are (2,2), (5,2), (5,5), (2,5) and the given viewport location is (0.5,0), (1,0), (1,0.5), (0.5,0) Calculate the viewing transformation matrix.
16. A window is defined by coordinates (0,30), (0,30) respectively and line with end points P1(-10,15) and P2(15,-10). Clip a line by using midpoint subdivision algorithm.
17. Explain the concept of Window and view-port.
18. What is windowing? State its utility in Graphics packages.
19. Compare and contrast between different clipping algorithms.
20. Define the following:
21. World co-ordinates
22. Device co-ordinates
23. Window co-ordinates
24. What is clipping? Explain cohen Sutherland line clipping.
25. Explain polygonal clipping with neat sketches.
26. Clip the polygon given in the figure using Sutherland Hodgeman polygon clipping algorithm.

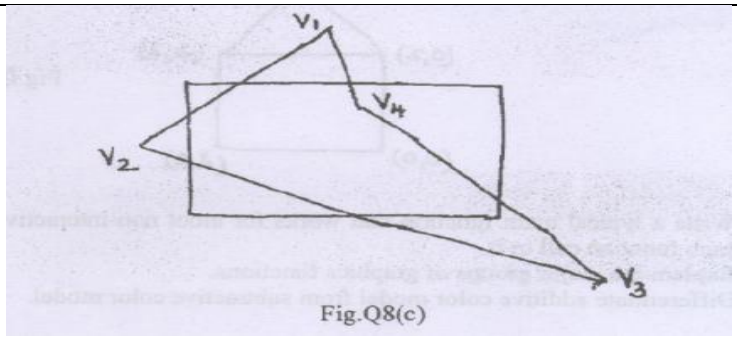


Fig.Q8(c)

27. Explain the Scan line fill algorithm.

UNIT-III

1. What is Transformation?
2. What is shearing?
3. What is reflection?
4. Distinguish between window port & view port.
5. Define clipping.
6. What is the need of homogeneous coordinates?
7. What is fixed point scaling?
8. List out the various Text clipping.
9. What is the use of clipping?
10. How will you clip a point?
11. Define viewing transformation.
12. What are the various representation schemes used in three dimensional objects?
13. What is Polygon mesh?
14. Define B-Spline curve.
15. What is a spline?
16. What is the use of control points?
17. What are the different ways of specifying spline curve?
18. What are the important properties of Bezier Curve?
19. Differentiate between interpolation spline and approximation spline.
20. Define Projection.
21. What do you mean by view plane?
22. What is view-plane normal vector?
23. Explain reflection and shear?
24. Explain Sutherland Hodgeman polygon clipping
25. Explain about clipping operations.
26. Find the transformation matrix that transforms the given square ABCD to half its size with center still remaining at same position. The co-ordinate of square are A(1,1) B(3,1) C(3,3) D(1,3) and center at (2,2). Also find resultant co-ordinate of square.
27. Explain in detail about window to viewport coordinate transformation.
28. Write a detailed note on the basic two dimensional transformations.
29. Explain spline representation

30. Explain Back face detection method and Depth buffer method
31. Explain area subdivision and A- Buffer method
32. Briefly explain about the basic transformations performed on three dimensional objects.
33. Write short notes on parallel and perspective projections.
34. Explain in detail about three dimensional display methods.
35. Explain in detail about the boundary representation of three dimensional objects.
36. Explain in detail about the three dimensional transformations.
37. Explain in detail about 3D window to viewport coordinate transformation.
38. What are Parallel and Perspective Projection? Explain with neat sketch 2-point and 3-point Perspective projection.
39. Obtain 3-D transformation matrix for
    a) Translation
    b) Scaling
    c) Rotation about Z-Axis.
40. Derive matrix for rotation about arbitrary point. Also rotate point (3, 3) with respect to (1, 1) by 90 degree.
41. Explain the following transformation with the matrix representations. Give suitable diagram for illustration
    - Reflection.
    - Shearing.
42. How the scaling of an object about the pivot point is performed?
43. Prove
    - Translations are additive.
    - Rotations are additive.
    - Scaling are multiplication.
44. Derive the transformation matrix for rotation about origin and rotation about fix point.
45. Rasterize the circle with radius r=5 and center = (100,100) with midpoint circle generation algorithm.
46. Rasterize the line from (10, 5) to (15, 9) using Bresenhams line drawing Algorithm.
47. Find the reflection of a triangle defined by the vertices A (1, 1), B (5, 1) and C (1, 5) about a line y=2x+10.
48. Derive transformation matrix for 3D rotation followed by translation followed by scaling 29.
49. Derive transformation matrix for rotation about arbitrary axis passing through origin
50. Explain With suitable diagram: Parallel projection
51. Explain With suitable diagram: Perspective projection.
52. Explain transformation in homogeneous coordinates.
53. Explain concatenation of transformations.
54. Describe instance transformation.
55. Explain rotation about arbitrary axis.
56. What is incremental rotation?
57. Explain the following in homogeneous

i) Rotation
ii) ii) Translation
iii) iii) Scaling
iv) iv) Shear

58. Explain the how rotation is made for the following when rotation is considered.
    i) About a fixed point
    ii) General rotation
    iii) About an arbitrary axis
59. Determine the rotation matrix formed for
    i) About a fixed point
    ii) General rotation
    iii) About an arbitrary axis.
60. Explain the basic transformations in 3D and represent them in matrix form.
61. What is axonometric projection?
62. What is isometric trimetric and diametric view?
63. Explain in detail about perspective projection.
64. Explain oblique projections.
65. . A unit square is transformed by 2*2 transformation matrix. The resulting position vectors are (0 2 8 6), (0 3 4 1).
66. What is the transformation matrix?
67. Find the matrix that represents rotation of an object by 45 about the origin.
68. What are the new co-ordinates of the point P (2,-4) after the rotation? Triangle is define by (2 4 4), (2 2 4).
69. Find the transformed co-ordinates after the following transformation
    (1) 90 rotation about origin.
    (2) Reflection about line y= -x.
70. Translate the square ABCD whose co-ordinate are A(0,0),B(3,0),C(3,3) and D(0,3) by 2 units in both directions and then scale it by 1.5 units in x-direction and 0.5 units in y-direction.
71. Q5.Perform a 45 rotation of triangle A(0,0),B(1,1),C(5,2)
    (a) About the origin and
    (b) About point P (-1,-1).
72. Use the Cohen-Sutherland algorithm to clip line P1(70,10) and P2(100,10) against a window lower left hand corner(50,10) and upper right hand corner(80,40)
73. Explain problems with Sutherland-Hodgeman algorithm? Also write at least two ways to solve these problems?
74. Perform a 450 rotation of triangle A(0,0),B(1,1),C(5,2) about the point P (-1,-1)?
75. Explain Weiler and Atherton polygon clipping algorithm? This algorithm is better or not than Sutherland Hodgeman polygon clipping algorithm give your answer?
76. Use Liang-Barsky line clipping algorithm to clip the line P1 (-15,-30) to P2 (30, 60) against the window having diagonally opposite corners as (0, 0) and (15, 15)?
77. What are the basic transformations? Describe each with their matrix representation.
78. What do you understand by homogenous co-ordinate?
79. Prove that two 2D transformation are commute i.e T 1T2=T2T1.

80. Show that the composition of two rotations is additive by concatenating the matrix representations for R(θ1) and R (θ2) to obtain R(θ1) R (θ2)= R(θ1+θ2).
81. Show the composition of two scaling is multiplicative by concatenating the matrix representation for [S(sx2,sy2).S(sx1,sy1)=S(sx1.sx2,sy1.sy2)]
82. What is shearing transformation? Explain with example.
83. Prove that two 2D rotation about origin; commute i.e R1R2=R2R1.
84. Show that a reflection about the line y=-x is equivalent to a reflection relative to y-axis followed by a counterclockwise rotation of 90.
85. Find the instance transformation which places a half –size copy of square A(0,0),B(2,0), C(2,2) and D(0,2) defined in a master co-ordinate system into a world co-ordinate system in such a way that the center of the square is at(-3,-3) in the world co-ordinate system.
86. Show the three transformation matrices used to rotate a figure 90 anti-clockwise about (3, 3).Also generate the transformation matrix.
87. Prove the simultaneous shearing in both directions(x and y) is not equal to the composition of pure shearing along x-axis followed by pure shear along y-axis.

UNIT-IV

1. Define complementary colors.
2. Define primary colors.
3. Define intensity of light.
4. What is hue?
5. What are the properties of light?
6. What is purity of light?
7. Define the term chromaticity.
8. State the use of chromaticity diagram.
9. Explain in detail about XYZ color model.
10. Explain in detail about RGB color model
11. Explain in detail about YIQ color model.
12. Explain in detail about CMY color model.
13. Explain in detail about HSV color model.
14. Compare and contrast RGB and CMY.
15. Explain in detail about the conversion between HSV and RGB color models.
16. Explain in detail about HLS color model.
17. Explain various methods to specify motion of object.
18. Explain RGB and HIS color model.
19. Enlist and explain any two color models.
20. Explain the features of computer graphics and animation software.
21. Explain a segment table with an example along with data structure used to implement are segment table?
22. Write any two algorithms for segment operation.
23. Explain methods of controlling animation.

24. Compare and contrast conventional and computer based animation.
25. What is segment table? Explain different operation on it.
26. Explain: RGB color model and CMY color model
27. State the difference between CMY and HSV color models.
28. What are the subtractive colors?
29. Define YIQ Color Model.

UNIT-V

1. What do you mean by shading of objects?
2. What are the types of reflection of incident light?
3. Define rendering.
4. Differentiate flat and smooth shading.
5. Define shading.
6. What is a shadow?
7. Explain in detail about shading models.
8. Explain about shading and graphics pipeline.
9. Compare Flat shading and Smooth shading.
10. Explain Gouraud shading and Phong shading
11. What is shading? Explain phong shading with its advantage and disadvantage.
12. Write short note on the following
    a) Backface removal algorithm.
    b) Painter's algorithm
    c) Z–buffer
13. Explain point source illumination and diffused illumination.
14. Write a note on Hidden surface removal.
15. Explain z-buffer algorithm with suitable example.
16. Write a short note on scan line algorithm.
17. Explain: A. Diffuse reflection, B. Specular reflection.
18. Write a note on shading algorithm.
19. What do you meant by shading of objects?
20. What are the types of reflection of incident light?
21. Define Rendering.
22. Differentiate flat and smooth shading.
23. What is meant by shadow?
24. What are the two methods for computing shadows? Explain in detail.
25. Write any two Drawbacks of Phong Shading.
26. What are the two common sources of textures? Explain in detail.
27. Write two types of smooth shading. Explain in detail.
28. What is a color model?
29. What is purity of light?
30. How is the color of an object determined?
31. What is meant by purity or saturation?
32. What is meant by complementary colors?

33. State the use of chromaticity diagram.
34. What is Color Look up table?
35. Explain in detail the basic illumination models.

UNIT-VI

1. Define graphics animation.
2. Define frame.
3. What is Fractals?
4. What is random fractal?
5. What is Koch curve?
6. Explain about fractals and self-similarity.
7. Give an account about Peano curves.
8. Give a detailed account of random fractals.
9. Explain how to find the intersection of a ray with an object.
10. Give an account on adding surface texture.
11. Explain about reflections and transparency.
12. Write short notes on :
    a. Bezier Curve
    b. B-splines
    c. Transparency
13. Discuss the properties of the Bezier and B-Spline curves?
14. Construct enough points on the Bezier curve whose control points are P0(4,2),P1(8,8),P2(16,4) to draw an accurate sketch
    - What is the degree of the curve?
    - What are the co-ordinates at $\mu = 0.5$?
15. Explain scan line algorithm for hidden surface removal?
16. Explain Back Face Removal algorithm for Hidden surface removal?
17. Differentiate between the object space and the image space hidden surface algorithm. .Explain the following:
    1)A-buffer
    2) Z-buffer
18. What is the difference between interpolation spline and approximation spline?
19. What do you understand by Bezier curve? A cubic curve is defined by the points (1,1) (2,3) (4,4) and (6,1).Calculate the co-ordinates of parametric midpoint of this curve ,and verify that its gradient dy/dx is 1/7 at this point. Sketch the curve.
20. What do you understand by blobby object and fractal objects?
21. What are peano curves?
22. What is purpose of rendering texture?
23. What is the use of fractals in graphics applications?
24. Define Fractals. Give examples.
25. List out some properties of fractal.

26. What are three types of self-similarity found in fractals?
27. What is Koch Curve?
28. Give the general procedure to construct Koch curve.
29. Draw the diagram of second generation of Koch snowflakes.
30. What is String Production Rules?

5 .Name of Course: Digital Electronics and Logic Design (210245)

Weekly Work Load (in Hrs)	Lecture	Tutorial	Practical
	03	-	02

Online/ In-sem	Theory	Practical	Oral	Term-work	Total Marks	Credit
30	70	-	-	25	125	04

Course Objectives

- To study number systems and develop skills for design and implementation of combinational logic circuits and sequential circuits
- To understand the functionalities, properties and applicability of Logic Families.
- To introduce programmable logic devices and ASM chart and synchronous state machines.
- To introduce students to basics of microprocessor

Course Outcomes

On completion of the course, learner will be able to ,

C205.1 Simplify Boolean Expressions using K Map.

C205.2 Design and implement combinational circuits.

C205.3 Design and implement sequential circuits.

C205.4 Develop simple real-world application using ASM and PLD.

C205.5 Differentiate and Choose appropriate logic families IC packages as per the given design specifications.

C205.6 Explain organization and architecture of computer system.

Syllabus

Unit	Course Contents	Hours
I	**Minimization Technique**	**06**
	Logic Design Minimization Technique -: Minimization of Boolean function using K-map(up to 4 variables) and Quine Mc-Clusky Method, Representation of signed number- sign magnitude representation ,1's complement and 2's complement form (red marked can be removed), Sum of product and Product of sum form, Minimization of SOP and POS using K-map.	
II	**Combinational Logic Design**	**06**
	Code converter -: BCD, Excess-3, Gray code, Binary Code. Half- Adder, Full Adder, Half Subtractor, Full Subtractor, Binary Adder (IC 7483), BCD adder, Look ahead carry generator, Multiplexers (MUX): MUX (IC 74153, 74151), Cascading multiplexers, Demultiplexers (DEMUX)- Decoder (IC 74138, IC 74154), Implementation of SOP and POS using MUX, DMUX, Comparators (2 bit), Parity generators and Checker.	
III	**Sequential Logic Design**	**06**
	Flip-Flop: SR, JK,D,T; Preset &Clear, Master Slave JK Flip Flops, Truth Tables and Excitation tables, Conversion from one type to another type of Flop Flop. Registers: SISO, SIPO, PISO, PIPO, Shift Registers, Bidirectional Shift Register, Ring Counter , Universal Shift Register Counters: Asynchronous Counter, Synchronous Counter, BCD Counter, Johnson Counter, Modulus of the counter (IC 7490),Synchronous Sequential Circuit Design :Models- Moore and Mealy, State diagram and State Table ,Design Procedure, Sequence Generator and detector.	
IV	**Algorithmic State Machines and Programmable Logic Devices**	**06**
	Algorithmic State Machines: Finite State Machines (FSM) and ASM, ASM charts, notations, construction of ASM chart and realization for sequential circuits. PLDS:PLD, ROM as PLD, Programmable Logic Array (PLA), Programmable Array Logic (PAL), Designing combinational circuits using PLDs.	
V	**Logic Families**	**06**
	Classification of logic families: Unipolar and Bipolar Logic Families, Characteristics of Digital ICs: Fan-in, Fan-out, Current and voltage parameters, Noise immunity, Propagation Delay, Power Dissipation, Figure of Merits, Operating Temperature Range, power supply requirements. Transistor-Transistor Logic: Operation of TTL NAND Gate (Two input),	

	TTL with active pull up, TTL with open collector output, Wired AND Connection, Tristate TTL Devices, TTL characteristics. CMOS: CMOS Inverter, CMOS characteristics, CMOS configurations- Wired Logic, Open drain outputs.	
VI	**Introduction to Computer Architecture**	**06**
	Introduction to Ideal Microprocessor – Data Bus, Address Bus, Control Bus. Microprocessor based Systems – Basic Operation, Microprocessor operation, Block Diagram of Microprocessor. Functional Units of Microprocessor – ALU using IC 74181, Basic Arithmetic operations using ALU IC 74181, 4-bit Multiplier circuit using ALU and shift registers. Memory Organization and Operations, digital circuit using decoder and registers for memory operations.	

Text Books

1. 1. Modern Digital Electronics by R.P.Jain, 4th Edition, ISBN 978-0-07-06691-16 Tata McGraw Hill
2. 2. Digital Logic and Computer Design by Moris Mano, Pearson , ISBN 978-93-325-4252-5

Reference Books

1. John Yarbrough, ―Digital Logic applications and Design, Cengage Learning, ISBN – 13: 978-81-315-0058-3
2. D. Leach, Malvino, Saha, ―Digital Principles and Applications‖, Tata McGraw Hill, ISBN 13:978-0-07-014170-4.
3. Anil Maini, ―Digital Electronics: Principles and Integrated Circuits‖, Wiley India Ltd, ISBN:978-
81-265-1466-3.
4. Norman B & Bradley, ―Digital Logic Design Principles, Wiley India Ltd, ISBN:978-81-265-1258-

Teaching plan

Sr. No.	Unit	Broad Topic to be covered	Total Lectures Planned
1	I	Minimization Technique	06
2	II	Combinational Logic Design	06
3	III	Sequential Logic Design	06
4	IV	Algorithmic State Machines and Programmable Logic Devices	06
5	V	Logic Families	06
6	VI	Introduction to Computer Architecture	06

Assignment Tool Details:

Sr. No.	Assessment Tool	Total in number	Marks scale down to
1.	MCQ Test	20 marks	20
2.	Simulation Based Assignment (SBA1)	20 marks	20
3.	Theory Tests	03 (each of 20 marks)	60
4.	Assignment (A1)	01	20
		Total	120

Assessment Tool Planner

Class – SE
Course Name – Digital Electronics and Logic Design
Course Code – (210245) **Course No.** – 203
Teaching Scheme **Marking Scheme**
Theory – 3Hrs/wk Theory Marks
MSE – 30**ESE** – 70

Sr. No.	Assessment Tool	Marks	Schedule
1	MCQ Test	20 marks	4th week of August 2021
2	Simulation Based Assignment (SBA1)	20 marks	4th week of Oct 2021
3	Theory Tests	03 (each of 20 marks)	Last week of Nov 2021
4	Assignment (A1)	15	2nd week of December 2021

Practical Assignment

Sr. No.	Title
LA1	To Realize Full Adder/ Subtractor using a) Basic Gates and b) Universal Gates
LA2	Design and implement Code Converters-Binary to Gray and BCD to Excess-3
LA3	Design and Realization of BCD Adder using 4-bit Binary Adder (IC 7483).
LA4	Realization of Boolean Expression for suitable combination logic using MUX 74151 /74153, DMUX 74154/74138
LA5	To Verify the truth table of two bit comparators using logic gates.
LA6	Design and Implement Parity Generator and checker using EX-OR.
LA7	Design and Realization: Flip Flop conversion
LA8	Design of 2 bit and 3 bit Ripple Counter using MS JK flip-flop
LA9	Design of Synchronous 3 bit Up and Down Counter using MSJK Flip Flop / D Flip Flop
LA10	Realization of Mod -N counter using (Decade Counter IC 7490) .
LA11	Design and implement Sequence generator (for Prime Number/odd and even) using MS JK flip-flop.
LA12	Design and implement Sequence detector using MS JK flip-flop.
LA13	Study of Shift Registers (SISO,SIPO, PISO, PIPO)
LA14	Design of ASM chart using MUX controller Method.

Unit wise Question bank

UNIT I Questions:

1. Explain functions of MUX & DEMUX.
2. Give IC number for the following:
    a. Quad 2:1mux
    1. Dual 4:1 mux
    2. 8:1 mux
    3. 16:1 mux
    4. Dual 1:4 demux
    5. 1:8 demux
    6. 1:16 demux
2. Which are the advantages of MUX?
3. Implement the following
    1. 8:1 mux using two 4:1 mux.
    2. 32:1 mux using two 16: 1 mux
    3. 16:1 mux using two 8:1 mux
4. What is MSB & LSB methods ?
5. Give applications of mux and demux.
6. What are codes? Give different types of codes.
7. Define weighted & Non-weighted codes.
8. Give applications of various codes.
9. Explain gray code and its applications.
10. List various types of codes with examples.
11. Explain how BCD addition takes place using 7483?
12. For which two conditions BCD correction is required?
13. Give applications of BCD code
14. Give features of IC 7483.
15. What is Parity?
16. What are different types of parity?
17. What is parity bit check?
18. Why it is necessary in digital communication?
19. How to generate even parity & odd Parity?

UNIT II Questions:

1. What is flip-flop? Give it's applications?
2. What are asynchronous & synchronous counter? Compare them.
3. Why asynchronous counters are also called as ripple counters?
4. Give application of counters.
5. What is lock-out condition in counter? How it can be avoided?
6. What is modulus of counter? What does it represent?
7. What is a sequence generator?
8. What are different types of flip-flops?
9. Convert the following :
10. S –R flip flop to J-K flip flop.

11. S –R flip flop to T flip flop.
12. S –R flip flop to D flip flop.
13. What is difference between combinational logic circuit & sequential logic circuit? Give example of each.
14. What is excitation table of flip-flop?
15. What is the difference between truth table & excitation table?
16. How race around condition is avoided in master slave J-K flip-flop
17. What are the applications of sequence generator?
18. What is mean by state diagram?
19. What do you mean by Sequence Detector?
20. What do you mean by Moore model and Mealy Model?
21. What are disadvantages of Mealy Model?
22. What is mode control input?
23. What is ring counter?
24. What is Johnson /Twisted ring counter?
25. Can J K flip-flop be converted to D type flip-flop? If yes how?
26. What is the basic use of IC 7490?
27. Explain internal design of 7490.
28. Explain functions of each pin in IC 7490.
29. What is the basic use of IC 74190?
30. How to cascade two 7490?
31. What is max/min pin? Explain Function.
32. What is load pin?
33. What is the basic use of IC 74194?
34. What is universal shift register?
35. Give applications of shift register.
36. What is a shift register? Explain its working.
37. Explain function table of 74194.

UNIT III Questions:

1. What is VHDL?
2. Explain the concept of data object
3. Describe the different modeling styles of VHDL with suitable examples.
4. Write a VHDL code for half adder using structural modeling
5. Write a VHDL code for 4 : 1 multiplexer.
6. In data flow modeling statements are executed ----------.
7. The statements with in PROCESS are executed ----------.
8. The signals written in parentheses after the keyword PROCESS is----------.
9. Write entity declaration for a 4 :1 mux
10. Explain the concept of PROCESS statement.

UNIT IV Questions:

1. What is PLD?
2. Explain PLA,PAL
3. What is macrocell?

4. What is the difference between PLA and PAL?
5. Explain AND matrix & OR matrix.
6. How to expand the capacity of PLA?
7. Give the applications of PLA
8. What is mean by SPLD & CPLD?
9. What is mean by FPGA?
10. Which are the logic families of FPGA?
11. What is Logic Cell Array?
12. Explain the concept of PLICE.
13. What is mean by Antifuse? State the difference between CPLD and FPGA

UNIT V Questions:

1. Name the different logic families. Which are the two commonly used families?
2. What are the various characteristics of digital IC's used to compare their performance? Define each of them.
3. Compare parameters of TTL & CMOS family.
4. Explain operation of TTL NAND gate, CMOS inverter, CMOS NAND and NOR gates.
5. Define the terms-Open collector o/p, unconnected inputs, open drain o/p.
6. Give significance of-74,74H,74KL,74S,74LS,74AS,74ALS,74HC.
7. What do you mean by interfacing?
8. Why TTL-CMOS & vice versa IC's cannot be interfaced directly?
9. What is TTL MOS?
10. Define tri-state logic.
11. Define various current and voltage parameters of logic gates.

UNIT VI Questions:

1. What is microcontroller? What are its types?
2. What is the difference between microcontroller and microprocessor?
3. Explain architecture of 8051 microcontroller in detail.
4. Explain pin diagram of 8051 microcontroller with diagram.
5. What are the applications of microcontroller?
6. What are the instruction sets of microcontroller?
7. What are the register sets of microcontroller?