N.C.College of Engineering

Israna-132107 (Panipat)

Scheme and Syllabus

w.e.f 2015-16 Session

Computer Science & Engineering

Second Year (3rd and 4th Semester)

MARKS DISTRIBUTION
(ACCORDING TO AUTONOMY)

FOR ALL THEORY COURSES: -

1. On Semester Evaluation of all theory courses
   Total: 100 Marks

Distribution

I. Mid Semester Examination	20 Marks	40 Marks	Best two will be included
II. Mid Semester Examination	20 Marks		
III. Mid Semester Examination	20 Marks		
Continuous Evaluation Test (CET)	20 Marks		
Attendance	20 Marks		
Teacher's Assessment	20 Marks		

2) End Semester (Final Examination) of all theory courses

Total: 100 Marks

3) Total of On Semester + End Semester Evaluation is of 200 Marks

4) To pass a theory course, the student should obtain

Minimum: - 80 Marks out of 200.

Criterion for passing and failing in the theory courses: -

a) The students will have to obtain 35% Marks in theory and 80 Marks in aggregate of On Semester and End Semester Evaluation for passing. If the above passing criterion is not fulfilled, the student will be awarded "Reappear".

b) On Semester Marks will not be changed. Only the theory marks will be modified as obtained in "Reappear".

c) If the attendance in a course is below 75%, the student will not be permitted to appear in the Final Examination.

FOR ALL PRACTICAL (LABORATORY) COURSES: -

I) On Semester Evaluation of all Practical (Laboratory) Courses

Total: 120 Marks.

Distribution

Attendance	60 Marks
Record of Practicals/ Experiments	30 Marks
Teacher's Assessment	30 Marks

II) End Semester Evaluation (Final Lab Examination + Oral Test or Viva Voce)

Total: 80 Marks

III) Total of On Semester Evaluation (Final Lab Examination) + End Semester Evaluation is of 200 Marks.

IV) To pass a lab course, the student should obtain

Minimum: 80 Marks out of 200.

Criterion for passing and failing in the lab course is just like the theory course.

CALCULATION OF SEMESTER GRADE POINT AVERAGE: -

Semester grade point average (SGPA) is the weighted average of the grade for the subjects registered **in a Semester** and is computed as follows:

$$SGPA = \frac{\sum_i C_i \times G_i}{\sum_i C_i}$$

C_i denotes the Credits (or Units) assigned to the ith subject and G_i denotes the Grade Point Equivalent to the Letter Grade obtained for the ith subject.

Cumulative Grade Point Average (CGPA) is the weighted average of the grades of the subjects for the registered in the semester.

N.C.COLLEGE OF ENGINEERING, ISRANA

SCHEME OF STUDIES AND EXAMINATION

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Computer Science & Engineering (Semester-III) 2015-19

Sr. No.	Subject Code	Subjects	BOS	L	T	P	Contact Hours	Credit
1	MGT-231	Industrial Economics	MGT	3	0	-	3	3
2	MATH-231	Mathematics-III	SCIENCE	3	1	-	4	4
3	MATH-200	Foundation Mathematics (for LEET Students only)	SCIENCE	2	0	-	2	2
4	CSE-231	Data Base Management Systems	CSE	3	1	-	4	4
5	CSE-232	Data Structures	CSE	3	1	-	4	4
6	ECE-233	Digital Electronics	ECE	3	1	-	4	4
7	CSE-233	Discrete Structures	CSE	3	1	-	4	4
8	CSE-23P1	DBMS Lab	CSE	-	-	2	2	1
9	CSE-23P2	Data Structures Lab	CSE	-	-	2	2	1
10	ECE-23P2	Digital Electronics Lab	ECE	-	-	2	2	1
11	CSE-23A	Society & School Connect Program	CSE					Audit Course
		Total		18/ 20	5	6	29/31	26/28

3rd Semester (CSE)

INDUSTRIAL ECONOMICS
MGT-231

L	T	
3	0	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

Note: **Examiner will be required to set NINE questions in all. Question Number 1 will consist of objective type/short-answer type questions covering the entire syllabus. In addition to this compulsory question there will be four units i.e. Unit-I to Unit-IV. Examiner will set two questions from each Unit of the syllabus. Each question will carry 20 marks.**

Student will be required to attempt FIVE questions in all. Question Number 1 will be compulsory. In addition to compulsory question, student will have to attempt four more questions selecting one question from each Unit.

UNIT-I

Introduction : Meaning, Nature and Scope of Economics, Correlation between Economics, Science, Engineering, Technology and Management. Managerial Economics and its scope in engineering perspective. Present condition of Indian Economics in the background of World Economy.

UNIT-II

Basic Macro Economic Concepts: GDP, GNP, National Income(NI), Business Cycles Concept of NI and Measurement. Inflation: Types, causes & prevention methods, Phases of business cycle, Reserve Bank of India and Its role in economic control, Concept of Currency and its control mechanism.

UNIT-III

Basic Concept of Investment: Primary Market, Secondary Market, Investors, Investment options, Issues of Securities, Types of securities, Regulatory Laws-SEBI ACT

UNIT-IV

Basic Concept of Corporate : Company and its different types, Non Government Organization (NGO), Consultancy Firm, Partnership Firm, Incorporation of company and general concept, Fund of a company: Its generation and operation, Share, Equity, Debenture, Bond, ESOP, Books of account.

Text Books:

1. Indian Economics, Dutt & Sundaram
2. A test book of Economic Theory by Stonier & Hague, Pearson
3. Indian Security Markets A Premier, By National Institute of Securities Markets, FPCIL

Reference Books:

4. Double entry book keeping by T.S. Grewal- S. Chand.
5. Modern Micro Economics by Theory - H.L.Ahuja-S.Chand
6. Managerial Economics for Engineering : Prof. D.N. Kakkar
7. Advance Economic Theory by M.L.Jhingan, Konark Publication

3rd Semester (CSE)

MATHEMATICS-III

MATH-231

L	T	W	On Semester Evaluation : 100 Marks
3	1	4	End Semester Evaluation : 100 Marks

Note: 1. There are nine questions in a set of question paper. All questions will carry equal marks.

2. The students are required to attempt five questions in all selecting one from each unit and First question is compulsory.

UNIT- I

Vector Calculus: Differentiation of vectors , scalar and vector point function Gradient of a scalar field and direction derivative , divergence and curl of a vector field and their physical interpretations, Integration of vectors, line integral, surface integral, Green, Stoke's and Gauss divergence theorems (without proof) and their simple applications.

Text Book: Grewal,B.S., 'Higher Engineering mathematics, Khanna Publishers (Chapter No.8) Exercise (8.3, 8.4, 8.5, 8.6, 8.7, 8.8, 8.9, 8.10)

UNIT-II

Partial Differential Equations: Formation of partial differential equations, Lagrange's linear partial differential equation, First order non-linear partial differential equation, Charpit's method. Method of separation of variables.Solution of one dimensional heat equation.

Text Book: Grewal,B.S., 'Higher Engineering mathematics, Khanna Publishers (Chapter 17,18, Exercise (17.1, 17.2, 17.3, 17.4, 17.5, 18.1, 18.3)

UNIT-III

Probability Distributions: Conditional Probability , Baye's theorem, Random variables Expectation , Discrete & Continuous probability distributions, Moment generating function,Probability generating function, Binomial, Poisson and Normal distributions.

Text Book: Grewal,B.S., 'Higher Engineering mathematics, Khanna Publishers (Chapter No26, Exercise (26.3, 26.4, 26.5, 26.6, 26.7)

UNIT-IV

Functions of Complex Variables: Exponential function, Trigonometric Differentiability and analyticity. Cauchy-Riemann equations, Necessary and sufficient conditions for a function to be analytic, Polar form of the Cauchy-Riemann equations, Harmonic functions, Orthogonality, Application to flow problems.

Complex Integration: Integration of complex functions, Cauchy-Integral theorem and formula,Taylor' and Laurent's series (without proof) ,Zeros and singularities of complex function, Residues , Cauchy Residue theorem.

Text Book: Grewal,B.S., 'Higher Engineering Mathematics, Khanna Publishers (Chapter No. 19,20,Exercise (19.5, 20.1, 20.5, 20.6, 20.7, 20.8)

Text Book:

1. Grewal,B.S., 'Higher Engineering Mathematice, Khanna Publishers.
2. Kresyzig, E.,"Advanced Engineering Mathematics" , John Wiley and Sons. (Latest edition).
3. Ramana ,B.V., 'Higher Engineering Mathematics" Tata McGraw-Hill.
4. Mathur A.B.,Jaggi V.P., "Advanced Engineering Mathematics", Khanna Publishers.
5. Babu Ram," Engineering Mathematics", Pearson Education.

Reference Book:

1. Jain, R.K. and lyengar, S.R.K.,"Advanced Engineering Mathematics", Narosa, 2003 (2^{nd}Ed.)
2. Goyal& Gupta, Function of complex variables" ,PragatiPrakashan
3. Probability and Statistics for Engineering :Johnson, Prentice Hall of India . Wylie, R., "Advanced Engineering Mathematics", McGraw-Hill, 1995.

3rd Semester (CSE)

FOUNDATION MATHEMATICS

(for LEET students)

MATH-200

L	T	W	
2	0	2	

On Semester Evaluation : 100 Marks

End Semester Evaluation : 100 Marks

Note: 1. There are nine questions in a set of question paper. All questions will carry equal marks.

2. The students are required to attempt five questions in all selecting at least one from each unit and First question is compulsory.

UNIT- I

Basics of Trigonometry: Definition of $\mathrm{Sin}\theta$, $\mathrm{Cos}\theta$ and $\tan\theta$. Law of Sines, Cosines and Tangents. Basic trigonometric identities, Sum and Difference formulas, Double angle and half angle formulas.

UNIT- II

Limit & Continuity of function of single variable, Fundamentals of differentiation, Algebra of differentiation, Differentiation of polynomials, trigonometric functions, logarithmic functions, exponential functions, hyperbolic functions and inverse functions.

UNIT- III

Differentiation of implicit and composite functions. Basics of partial differentiation. Formation of differential equation, Solution of differential equation: variable separable, Leibnitz linear equation, higher order differential equations.

UNIT- IV

Fundamentals of integration, Integration of polynomials, trigonometric functions, logarithmic functions, exponential functions, hyperbolic functions and inverse functions. Partial fraction, Integration by parts.

Text Books:

1. Elements of Business Mathematics, Jeevansons publications
2. Elementary Engineering Mathematics : B. S. Grewal, Khanna Publishers.
3. Applied Mathematics : R.D. Sharma Vol 1. Dhanpat Rai Publications.

Reference Books:

1. Differential Calculus by Shanti Narayan
2. Differential and Integral Calculus by N. Piskunov
3. Integral Calculus by Shanti Narayan

3rd Semester (CSE)

DATA BASE MANAGEMENT SYSTEMS
CSE-231

L	T	
3	1	**On Semester Evaluation : 100 Marks**
		End Semester Evaluation : 100 Marks

Note: 1. There are nine questions in a set of question paper. All questions will carry equal marks.

2. The students are required to attempt five questions in all selecting one from each unit and First question is compulsory.

UNIT-I

Basic Concepts: What is database system, why database, Data independence, 3 levels of architecture; external level, conceptual level, internal level, mapping DBA, DBMS, organization of databases, components of DBMS, Data Models, Relational Models, Networks data model, Hierarchical Model, semantic data model.

UNIT-II

Relational Model: Introduction – Relational Model, base tables & views, relations, domains, candidate keys, primary key, alternate keys, foreign key, Integrity rules, relational Operators - relational algebra, relational calculus, Data Base Design – Introduction, Basic Definitions, Non-loss decomposition and functional dependencies, 1NF, 2NF, 3NF, BCNF, MVD & 4NF, JD & 5NF, Normalization procedure, other normal forms.

UNIT-III

Concurrency: Transaction concept, transaction state, concurrent executions, serializability lock based protocols, timestamp based protocols, validation based protocols, deadlock handling.

UNIT- IV

Distributed Data Bases: Introduction, fundamental principles, objectives, Problems of distributed processing-query processing, catalog management, updates propagation, recovery control, and concurrency control.

Text Books:

1. Database System Concepts by A. Silberschatz, H.F. Korth and S. Sudarshan, 3rd edition, 1997, McGraw-Hill, International Edition.
2. Introduction to Database Management system by Bipin Desai, 1991, Galgotia Pub.

Reference Books:

1. Fundamentals of Database Systems by R. Elmasri and S.B. Navathe, 3rd edition, 2000, Addision-Wesley, Low Priced Edition.
2. An Introduction to Database Systems by C.J. Date, 7th edition, Addison-Wesley, Low Priced Edition, 2000.
3. Database Management and Design by G.W. Hansen and J.V. Hansen, 2nd edition, 1999, Prentice-Hall of India, Eastern Economy Edition.
4. Database Management Systems by A.K. Majumdar and P. Bhattacharyya, 5th edition, 1999, Tata McGraw-Hill Publishing.

3rd Semester (CSE)

DATA STRUCTURES

CSE-232

L	T	
3	**1**	**On Semester Evaluation : 100 Marks**
		End Semester Evaluation : 100 Marks

Note: 1. There are nine questions in a set of question paper. All questions will carry equal marks.

2. The students are required to attempt five questions in all selecting one from each unit and First question is compulsory.

UNIT-I

INTRODUCTION: Introduction to Data Structures: Definition & Abstract data types, Static and Dynamic implementations

Order analysis: Objectives of time and space algorithms, Big O Notation and theta notations

Arrays: Definition, Implementation, lower bound, upper bound, addressing an element at a particular index for one dimensional arrays, Two dimensional arrays and Multi-dimensional arrays.

SORTING AND SEARCHING: Searching: Linear search, Binary search. Sorting: Insertion sort, selection sort, bubble sort, merge sort, heap sort

UNIT-II

ELEMENTARY DATA STRUCTURES: Stacks & Queues, Infix conversions, evaluations of expressions, stacks and queues, priority queues as heaps, deques, Implementation and applications of Stacks and Queues

LINKED LISTS: Singly linked lists, linked stacks and queues, polynomial addition, sparse matrices, doubly linked lists and dynamic storage management, Applications of Linked Lists, garbage collection, Josephus Problem.

UNIT-III

TREES: Basic terminology, binary trees, binary tree traversal, application of trees, set representation, decision tree, Binary Search Trees Tree, Height Balanced tree (AVL) and various rotations

UNIT-IV

GRAPH THEORY: Graph representations, Graph Traversals:BFT & DFT, Dijkstra's algorithm for shortest path, Spanning trees , Prim Algorithm, Kruskal Algorithm

TABLES: Definition, Hash Functions, Implementation & Applications

Text Book:

1. Data Structures using C by A. M. Tenanbaum, Langsam, Moshe J. Augentem, PHI Pub.

Reference Books:

1. Data Structures and Algorithms by A.V. Aho, J.E. Hopcroft and T.D. Ullman, Original edition, Addison-Wesley, 1999, Low Priced Edition.
2. Fundamentals of Data structures by Ellis Horowitz & Sartaj Sahni, Pub, 1983,AW
3. Fundamentals of computer algorithms by Horowitz Sahni and Rajasekaran.
4. Data Structures and Program Design in C By Robert Kruse, PHI,

5. Theory & Problems of Data Structures by Jr. Symour Lipschetz, Schaum's outline by TMH
6. Introduction to Computers Science -An algorithms approach , Jean Paul Tremblay, Richard B. Bunt, 2002, T.M.H.
7. Data Structure and the Standard Template library – Willam J. Collins, 2003, T.M.H

3rd Semester (CSE)

DIGITAL ELECTRONICS
ECE-233

L	T	
3	1	**On Semester Evaluation : 100 Marks**
		End Semester Evaluation : 100 Marks

Note:
1. **There are nine questions in a set of question paper. All questions will carry equal marks.**
2. **The students are required to attempt five questions in all selecting one from each unit and First question is compulsory.**
3. **The question paper will have 25% numerical part.**

UNIT-I

Fundamentals of Digital Techniques: Digital signal, Comparison of analog & digital systems, Logic gates : AND, OR, NOT, NAND, NOR, EX-OR, EX-NOR, Boolean algebra, Binary codes: BCD, Excess-3, Gray codes, ASCII codes.

Combinational Design Using Gates: Standard representation for logical functions, Design using gates, Karnaugh map and Quine-Mcluskey methods of simplification up to six variables.

UNIT-II

Combinational Design Using Mst Devices: Adders:-Half Adder, Full Adder(Serial & Parallel), Look Ahead Carry, Subtracters: Half Subrtacter,Full Subtracter, BCD arithmetic circuits, Comparators, Multiplexers and Demultiplexers and their use as logic elements, Encoders, Decoders, BCD to seven segment display devices.

Sequential Circuits: Flip Flops: S-R, D, J-K,T, Master-slave, Edge-triggered & level-triggered flip-flops, Conversion of flip-flops, Shift registers, Counters : Asynchronous and Synchronous, Ring counters and Johnson Counters, Design of Synchronous and Asynchronous sequential circuits.

UNIT-III

Digital Logic Families: Switching Characteristics of diodes and transistors, Characteristics of Digital ICs, Bipolar logic families: RTL, DTL, DCTL. HTL, TTL (Totem pole, schottky arrangement), ECL, MOS and CMOS logic families, Interfacing CMOS & TTL, Tristate logic.

UNIT-IV

A/D and D/A Converters: Sample and hold circuit, Quantization, D/A converters :- Weighted Resistor and R -2 R ladder D/A Converters, Specifications for D/A converters, A/D converters:- Parallel-comparator, Successive approximation, Dual-slope ADC, Specifications of ADCs.

Programmable Logic Devices: ROM, PROM, EPROM, EEPROM, Flash Type, RAM: Static and Dynamic memory, PLA. PAL, Introduction to FPGA and CPLDs.

Text Books:

1. Modern Digital Electronics (Edition III): R. P. Jain; TMH
2. Digital Principles and Applications: Malvino & Leech; McGraw Hill.

Reference Books:

1. Digital Integrated Electronics: Taub & Schilling: MGH
2. Fundamentals of Digital Circuits: Anand Kumar; PHI
3. Digital Design: Morris Mano: PHI

3rd Semester (CSE)

DISCRETE STRUCTURES
CSE-233

L	T	
3	1	**On Semester Evaluation : 100 Marks**
		End Semester Evaluation : 100 Marks

Note: 1. There are nine questions in a set of question paper. All questions will carry equal marks.

2. The students are required to attempt five questions in all selecting one from each unit and First question is compulsory.

UNIT-I

Sets and Propositions: Introduction, Combination of sets, Finite and Infinite sets, Uncountably Infinite Sets, Mathematical Induction, Principle of Inclusion and Exclusion, Multisets, Properties of Binary Relations, Equivalence Relations and Partitions, Partial Ordering Relations, Functions and Pigeonhole Principle, Propositions.

UNIT-II

Algebraic System: Definitions and elementary properties of algebraic structures, Semigroups, monoids and submonoids, Groups and subgroups, Homomorphisms and Isomorphisms of Monoids and Groups, Definition and Examples of Rings and Subrings, Types of Rings, Commutative Ring, Ring with Unity, Ring with or without Zero divisions, Integral Domain, Division Ring, Relation of Isomorphism in the set of rings, Field, its characteristics and subfield.

UNIT-III

Graphs and Planar Graphs: Introduction, Basic Terminology, Multigraphs and Weighted Graphs, Paths and Circuits, Shortest Paths in Weighted Graphs, Eulerian Paths and Circuits, Hamiltonian Paths and Circuits, Planar Graphs, Trees, Rooted Trees, Path Lengths in Rooted Trees, Binary Search Trees, Spanning Trees and Cut-sets, Minimum Spanning Trees.

UNIT-IV

Permutations, Combinations and Recurrence Relations: The Rules of Sum and Product, Permutations, Combinations, Generation of Permutations and Combinations, Recurrence Relations, Linear Recurrence Relations with Constant Coefficients, Homogeneous Solutions, Particular Solutions, Total Solutions, Solution by the Method of Generating Functions.

Note: The examiner will set nine questions, taking first question from entire syllabus, and two questions from each unit. Students are required to attempt five questions in all selecting at least one question from each unit and first question is compulsory. All questions will carry equal marks.

Text Book:

1. Elements of Discrete Mathematics C.L Liu, 1985, McGraw Hill

Reference Books:

1 Concrete Mathematics: A Foundation for Computer Science, Ronald Graham, Donald Knuth and Oren Patashik, 1989, Addison-Wesley.
2. Mathematical Structures for Computer Science, Judith L. Gersting, 1993, Computer Science Press.
3. Applied Discrete Structures for Computer Science, Doerr and Levasseur, (Chicago: 1985,SRA)
4. Schaums Outline series: Theory and problems of Probability by S. Lipshutz, 1982, McGraw-Hill Singapore

5. Discrete Mathematical Structures, B. Kolman and R.C. Busby, 1996, PHI
6. Discrete Mathematical Structures with Applications to Computers by Tembley & Manohar, 1995, Mc Graw Hill.

3rd Semester (CSE)

DATA BASE MANAGEMENT SYSTEMS LAB (DBMS LAB)

CSE-23P1

P **On Semester Evaluation: 120**
2 **End Semester Evaluation: 80**

LIST OF EXPERIMENTS:

1. Write the queries using Data Definition Language commands such as Create, Alter, trunc, rename, drop etc. using constraints (Primary key, Unique, Check, Not Null and Foreign Key).

2. Write queries using Data Manipulation Language commands such as Insert, Delete, Update, merge etc.

3. Write SQL queries to retrieve the data (Select Clause, Where clause etc.) from the database using Arithmetic operators (+, -, /, * etc.), Logical operators (AND, OR, NOT), Comparison operators (=, <, >, etc.), SQL operators (Between…. AND, IN (List), Like, IS NULL, ||,), column aliases, Order by clause etc.

4. Write SQL query using character, number, date, conversion, and general functions.

5. Write SQL queries for extracting data from more than one table using various joins available in SQL.

6. Write SQL queries using group functions, group by clause, sub queries and nested queries.

7. Write SQL queries to implement the concepts for ROLLBACK, COMMIT & SAVEPOINTS.

8. Create views, Index and synonyms for the user tables.

9. Write a program in PL/SQL to implement control statements.

10. Write a program in PL/SQL to implement Exception Handling concept.

11. Write a program in PL/SQL to implement procedure and function.

12. Write a program to create DML triggers on the user tables, Execute the triggers and drop them.

3rd Semester (CSE)

DATA STRUCTURES LAB

CSE-23P2

P	**On Semester Evaluation: 120**
2	**End Semester Evaluation: 80**

LIST OF EXPERIMENTS:

1. Write a program to search an element in two-dimensional array using linear search.
2. Using iteration and recursion concepts write a program for finding an element in the array using binary search method.
3. Write a program to perform following operation tables using functions only
    a) Addition b) Subtraction c) Multiplication d) Transpose.
4. Write a program to implement queues.
5. Write a program to implement stack.
6. Write a program to implement various operations on string such as length of string, concatenation, reverse and copy of string to another.
7. Write a program for swapping of two numbers using call-by-value and call-by-reference strategies.
8. Write a program to implement binary search trees
9. Write a program to make a copy of a given list.
10. Write a program to determine whether two lists are identical or not.
11. Write a program to create a linked list and perform operations such as insert, delete, update and reverse the linked list.
12. Write a program that deletes the kth element from a two-way circular header list.
13. Write a program for implementation of a file and performing operations such as insert, delete, update a record in file.
14. Create a linked list and perform following operations:
    a) Add a node
    b) Delete a node
15. Write a program to simulate various searching and sorting algorithms and compare their timings for a list of 1000 elements.
16. Write a program to simulate various graph traversing algorithms.
17. Write a program which simulates various tree traversal algorithms
18. Write a program for deleting a node from binary search tree.
19. Write a program to implement various searching techniques
20. Write a program to implement various sorting techniques.

3rd Semester (CSE)

DIGITAL ELECTRONICS LAB
ECE-23P2

P	**On Semester Evaluation: 120**
2	**End Semester Evaluation: 80**

LIST OF EXPERIMENTS: (At least 10 experiments are to be performed from list below**)**

1. Verification of TTL gates AND, OR, NOT, NAND, NOR, EX-OR, EX-NOR.
2. Design and realize a given function using K-Maps and verify its performance.
3. To verify the operation of half adder and full adder using gates.
4. To verify the operation of half subtractor and full subtractor using gates.
5. To verify the operation of Comparator.
6. To verify the operation of Multiplexer and Demultiplexer.
7. To verify the operation of Encoder and Decoder.
8. To verify the operation of BCD to 7 segment Decoder.
9. To verify the truth table of S-R, J-K, T, D Flip-flops.
10. To verify the operation of Bi-directional shift register.
11. To design and verify the operation of decade counter.

N.C.COLLEGE OF ENGINEERING, ISRANA

SCHEME OF STUDIES AND EXAMINATION

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Computer Science & Engineering (Semester-IV) 2015-19

Sr. No.	Subject Code	Subjects	BOS	L	T	P	Contact Hours	Credit
1	CSE-241	Object Oriented Programming using C++	CSE	3	1	-	4	4
2	CSE-242	Computer Organization & Architecture	CSE	3	1	-	4	4
3	CSE-243	Software Engineering	CSE	3	1	-	4	4
4	CSE-244	Operating Systems	CSE	3	1	-	4	4
5	ECE-354	Microprocessor and Interfacing	ECE	3	1	-	4	4
6	CSE-24P1	OOPs using C++ Programming Lab	CSE	-	-	2	2	1
7	CSE-24P2	Software Engineering Lab	CSE	-	-	2	2	1
8	ECE-35P1	Microprocessor and Interfacing Lab	ECE	-	-	2	2	1
9	CSE-24P3	Operating Systems Lab	CSE	-	-	2	2	1
10	CSE-24P4	Technical Seminar	CSE	-	-	2	2	1
11	CSE-245	General Proficiency and fitness	CSE	-	-	-	-	1
		Total		15	5	10	30	26

4th Semester (CSE)
OBJECT ORIENTED PROGRAMMING USING C++
CSE-241

L	T	
3	1	**On Semester Evaluation : 100 Marks**
		End Semester Evaluation : 100 Marks

Note: 1. There are nine questions in a set of question paper. All questions will carry equal marks.

2. The students are required to attempt five questions in all selecting one from each unit and First question is compulsory.

UNIT-I

Introduction to C++, differences between procedure oriented & object oriented programming, C++ Standard Library, Basics of a Typical C++ Environment, Header Files and Namespaces. Object Oriented Concepts: Introduction to Objects and classes, Encapsulation & Abstraction, Access Modifiers: Controlling access to a class, method, or variable (public, protected, private), Polymorphism, Overloading, Inheritance, Overriding Methods, Dynamic Binding, Message Passing. Benefits & Applications of OOP.

UNIT-II

Classes and Data Abstraction: Introduction, Structure Definitions, Accessing Members of Structures, Class Scope and Accessing Class Members, Dynamic Initialization of Variables, Reference Variables, Initializing Class Objects, Constructors, Default Arguments With Constructors, Copy Constructors, Destructors, Classes: Const (Constant) Object And Const Member Functions, Object as Member of Classes, Friend Function and Friend Classes, Dynamic Memory Allocation with New and Delete, Static Class Members, Container Classes.

Overloading: Function overloading, Operator Overloading, Restrictions On Operators Overloading, Operator Functions as Class Members v/s as Friend Functions, Overloading, <<, >> Overloading Unary Operators, Overloading Binary Operators.

UNIT-III

Inheritance: Introduction, Inheritance: Base Classes and Derived Classes, Protected Members, Casting Base- Class Pointers to Derived- Class Pointers, Using Member Functions, Overriding Base –Class Members in a Derived Class, Public, Protected and Private Inheritance, Using Constructors and Destructors in derived Classes, Implicit Derived –Class Object To Base- Class Object Conversion, Composition Vs. Inheritance.

Virtual Functions and Polymorphism: Introduction to Virtual Functions, Using This Pointer, Abstract Base Classes And Concrete Classes, Virtual Destructors, Dynamic Binding.

UNIT-IV

Files and I/O Streams: Files and Streams, Creating a Sequential Access File, Reading Data From A Sequential Access File, Updating Sequential Access Files, Random Access Files, Creating A Random Access File, Writing Data Randomly To a Random Access File, Reading Data Sequentially from a Random Access File. Stream Input/Output Classes and Objects, Stream Output, Stream Input, Unformatted I/O (with read and write), Stream Manipulators, Stream Format States, Stream Error States.

Templates & Exception Handling: Function Templates, Overloading Template Functions, Class Template, Class Templates and Non-Type Parameters, Templates and Inheritance, Templates and Friends, Templates and Static Members, Basics of C++ Exception Handling: Try Throw, Catch,

Throwing an Exception, Catching an Exception, Rethrowing an Exception, Exception specifications, Processing Unexpected Exceptions, Stack Unwinding, Exceptions and Inheritance.

Note: The examiner will set nine questions, taking first question from entire syllabus, and two questions from each unit. Students are required to attempt five questions in all selecting at least one question from each unit and first question is compulsory. All questions will carry equal marks.

Text Books:

1. Object oriented Programming with C++ by E Balagurusamy, 2001, Tata McGraw-Hill
2. C++ How to Program by H M Deitel and P J Deitel, 1998, Prentice Hall

Reference Books:

1. Object Oriented Programming in Turbo C++ by Robert Lafore, 1994, The WAITE Group Press.
2. Programming with C++ By D Ravichandran, 2003, T.M.H
3. Computing Concepts with C++ Essentials by Horstmann, 2003, John Wiley,
4. The Complete Reference in C++ By Herbert Schildt, 2002, TMH.

4th Semester (CSE)

COMPUTER ORGANIZATION & ARCHITECTURE

CSE-242

L	T	
3	1	**On Semester Evaluation : 100 Marks**
		End Semester Evaluation : 100 Marks

Note: 1. There are nine questions in a set of question paper. All questions will carry equal marks.

2. The students are required to attempt five questions in all selecting at least one from each unit and First question is compulsory.

UNIT-I

Basic Machine Principle, Structure and representation of real world data, Von-Newman Model and stored program concept, Subroutine, Branching & Macro facility, Processor Organization, Instruction cycle and Instruction format, Addressing modes, Arithmetic operation, Fixed and Floating Point arithmetic: addition, subtraction, multiplication and division, ALU design,

UNIT-II

Parallel processing – Performance consideration, Pipeline processor and Multiunit processor, Instruction sequencing and Interpretation, Hardware Control design method, Multiplier control unit and CPU control unit, Micro programmed Control, Minimizing Instruction Size, Micro programmed computer.

UNIT-III

Memory device characteristic, Random access and serial access memories, Virtual memory-memory hierarchies, Main Memory allocation & replacement policies, Segments, pages and file organization, High speed memories – Interlocked, cache and associative memory.

UNIT-IV

Programmed I/O, DMA and interrupts, I/O processors & CPU – I/O interaction, Types of instruction: Memory Reference, Register Reference and I/O Reference.

Text Books:

1. Computer Organization and Design, 2nd Ed., by David A. Patterson and John L. Hennessy, Morgan 1997, Kauffmann.
2. Computer Architecture and Organization, 3rd Edi, by John P. Hayes, 1998, TMH.

Reference Books:

1. Operating Systems Internals and Design Principles by William Stallings,4th edition, 2001, Prentice-Hall Upper Saddle River, New Jersey
2. Computer Organization, 5th Edi, by Carl Hamacher, Zvonko Vranesic,2002, Safwat Zaky.
3. Structured Computer Organisation by A.S. Tanenbaum, 4th edition, Prentice-Hall of India, 1999, Eastern Economic Edition.
4. Computer Organisation & Architecture: Designing for performance by W. Stallings, 4th edition, 1996, Prentice-Hall International edition.

4th Semester (CSE)

SOFTWARE ENGINEERING
CSE-243

L	T	
3	1	**On Semester Evaluation : 100 Marks**
		End Semester Evaluation : 100 Marks

Note: 1. There are nine questions in a set of question paper. All questions will carry equal marks.

2. The students are required to attempt five questions in all selecting at least one from each unit and First question is compulsory.

UNIT-I

Introduction to Software crisis & Software processes; Software life cycle models – Build & Fix, waterfall prototype evolutionary, spiral model, planning, scheduling staffing.

UNIT-II

Problem Analysis – DFD, Data dictionaries, ER diagrams, object diagrams; approaches to problems analysis; SRS; specifying behavioral & non-behavioral requirements, Modularity, strategy of design, function oriented design, object oriented design.

UNIT-III

Software Metrics: Introduction, size metrics, data structure metrics, information flow metrics, entropy-based measures, metric analysis, Software reliability & Hardware reliability, failures & faults, reliability concepts, Introduction to Reliability models. Software Quality, ISO,SIX SIGMA.

UNIT-IV

Software Testing: Introduction, Functional testing, structural testing, activities during testing, debugging, testing tools, Software Maintenance: Introduction, types of maintenance, maintenance process, maintenance models, reverse engineering, re-engineering.

Books:

1. Pressman S.Rogcr, Software Engineering. 'Tata McGraw-Hill
2. Jalote Pankaj, An integrated approach to software engineering .Narosa Publishing House
3. Sommerville Ian. Software Engineering. 5th ed., Addison Wesley-2000
4. Fairley Richard, Software. Software Engineering Concepts. Tata Mcgraw-Hill
5. Nasib Singh Gill, Software Engineering: Software Reliability , Testing and Quality Assurance , Khanna Book Publishing Co. (P) Ltd , New Delhi .

4th Semester (CSE)

OPERATING SYSTEM

CSE-244

L	T	
3	**1**	**On Semester Evaluation: 100 Marks** **End Semester Evaluation: 100 Marks**

Note: - 1. There are NINE questions in a set of question-paper. All questions carry equal marks.

2. Attempt five questions in all. FIRST question is compulsory which covers the whole syllabus. Attempt ONE question from each of the other four Units.

UNIT-I

Introduction and File Systems: Structure of Operating Systems Operating system functions and characteristics, historical evaluation of operating system, Multiprocessor Systems, Real time systems, Distributed systems, system programs, interrupt mechanisms.

File System: Functions of the systems, file access and allocation methods, Directory system: structured organization, Directory and file protection mechanism. Case study of File system in Linux/Unix

UNIT-II

Process Management: Process Concept – Process Scheduling – Operations on Processes – Cooperating Processes – Inter-process Communication.

CPU Scheduling: Levels of scheduling, comparative study of scheduling algorithms, multiple processor scheduling.

Concurrent Processes: Critical section problem, Semaphores, Classical process coordination, problems and their solutions, interprocess communication, multithreading.

UNIT-III

Memory and Device Management: Storage allocation methods: single contiguous allocation, multiple contiguous allocation, Paging, Segmentation, Combination of Paging and Segmentation, Virtual memory concepts, Demand paging, Page replacement algorithms, Thrashing.

Device Management: Hardware organization, device scheduling, policies and I/O Management.

Case Study of Memory management in Linux/Unix

UNIT-IV

Deadlocks and Protection:

Deadlock: Deadlock characterization, Deadlock prevention and avoidance, Deadlock detection and recovery, practical considerations.

Protection: Mechanism and Policies, implementation.

Books Recommended:

Text Book:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, "Operating System Concepts", Sixth Edition, John Wiley & Sons (ASIA) Pvt. Ltd, 2003.

Reference Book

1. Peterson,J.L.&Silbersehatz.A,Operating System concepts,Addison,Wesley.
2. Brinneh,Hansen,operating system principles:Prentiee Hall of India
3. Haberman,A.N.introduction to operating system,design Galgotia Publication, New Delhi.
4. Tanenbaum,A.N.:introduction to operating system

5. Hansen,P.B.Architecture of concurrent programs.PHI
6. Shaw,A.C.,Logic design of operating systems,PHI.

4th Semester (CSE)

MICROPROCESSORS & INTERFACING

ECE-354

L	**T**	**On Semester Evaluation: 100 Marks**
3	**1**	**End Semester Evaluation: 100 Marks**

Note: -

1. **There are nine questions in a set of question paper. All questions will carry equal marks.**
2. **The students are required to attempt five questions in all selecting one from each unit and First question is compulsory.**
3. **The question paper will have 25% numerical part.**

UNIT-I

Introduction: Evolution of microprocessors, CISC Versus RISC, Applications of microprocessors.

8085 CPU Architecture: 8085 programming model, 8085 hardware model, Microprocessor operation, 8085 architecture, 8086 pin diagram description, Bus timings: Opcode fetch, Memory read, Memory write .

8085 Instruction Set: Instruction formats, Addressing modes, Data Transfer instructions: 8 bit,16 bit; Arithmetic instructions: 8bit, 16 bit; Logic & bit manipulation instructions, Branch instructions, Machine control instructions, Stack, Subroutine, Programming examples.

UNIT-II

8086 CPU Architecture: 8085 versus 8086, 8086 block diagram: BIU and EU, Physical address computations, PSW with examples, 8086 pin diagram descriptions, Generating 8086 CLK and Reset signals using 8284, 8086 minimum mode and maximum mode CPU module.

8086 Instruction Set: Addressing modes, Data transfer instructions, String instructions, Logical instructions, Arithmetic instructions, Transfer of control instructions, Process control instructions, Assembler directives.

8086 Programming Techniques: Writing assembly language programs, Timing delays, Loops, Data conversions, Procedures, Modular programming, Macros.

UNIT-III

Interfacing Device: 8255 PPI Chip: Architecture, Pin diagram, Control words, Modes.

DMA: DMA operation, Intel 8237: Architecture, Working.

Interrupt: Interrupt driven I/O, Interrupt mechanism, Interrupt vector table of 8086, Interrupt structure of 8085.

Interrupt Controller: 8259: Architecture, Pin diagram, Working, ICWs, OCWs.

UNIT-IV

Main Memory System Design of 8085 & 8086: Memory devices, 8086 CPU Read/Write timing diagrams in minimum mode and maximum mode, Address decoding techniques, Interfacing SRAMS, ROMS/PROMS, Interfacing and refreshing DRAMs.

Basic I/O Interface: Memory mapped I/O vs Isolated I/O, Interfacing 8259, ADCs, DACs, Keyboards, Multiplexed displays with 8085 & 8086 microprocessors.

Reference Books:

1. D.V.Hall: Microprocessors and Interfacing, McGraw Hill (2nd Ed.)
2. J. Uffenbeck: The 8086/ 8088 family, PHI.
3. Liu, Gibson: Microcomputer Systems – The 8086/8088 family, PHI (2nd Ed.).
4. Ramesh S. Gaonkar, Microprocessor Architecture, Programming & Applications with 8085, Penram International Publications India Pvt. Ltd. (5th Ed.)

4th Semester (CSE)
OOPS USING C++ PROGRAMMING LAB
CSE-24P1

P **On Semester Evaluation: 120**
2 **End Semester Evaluation: 80**

LIST OF EXPERIMENTS:

1. Raising a number n to a power p is the same as multiplying n by itself p times. Write a function called power ( ) that takes a double value for n and an int value for p, and returns the result as double value. Use a default argument of 2 for p, so that if this argument is omitted, the number will be squared. Write a main ( ) function that gets values from the user to test this function.

2. A point on the two dimensional plane can be represented by two numbers: an X coordinate and a Y coordinate. For example, (4,5) represents a point 4 units to the right of the origin along the X axis and 5 units up the Y axis. The sum of two points can be defined as a new point whose X coordinate is the sum of the X coordinates of the points and whose Y coordinate is the sum of their Y coordinates.
   Write a program that uses a structure called point to model a point. Define three points, and have the user input values to two of them. Than set the third point equal to the sum of the other two, and display the value of the new point. Interaction with the program might look like this:
   Enter coordinates for P1: 3 4
   Enter coordinates for P2: 5 7
   Coordinates of P1 + P2 are : 8, 11

3. Create the equivalent of a four function calculator. The program should request the user to enter a number, an operator, and another number. It should then carry out the specified arithmetical operation: adding, subtracting, multiplying, or dividing the two numbers. (It should use a switch statement to select the operation). Finally it should display the result.
   When it finishes the calculation, the program should ask if the user wants to do another calculation. The response can be 'Y' or 'N'. Some sample interaction with the program might look like this.
   Enter first number, operator, second number: 10/ 3
   Answer = 3.333333
   Do another (Y/ N)? Y
   Enter first number, operator, second number 12 + 100
   Answer = 112
   Do another (Y/ N) ? N

4. A phone number, such as (212) 767-8900, can be thought of as having three parts: the a area code (212), the exchange (767) and the number (8900). Write a program that uses a structure to store these three parts of a phone number separately. Call the structure phone. Create two structure variables of type phone. Initialize one, and have the user input a number for the other one. Then display both numbers. The interchange might look like this:
   Enter your area code, exchange, and number: 415 555 1212
   My number is (212) 767-8900
   Your number is (415) 555-1212

5. Create two classes DM and DB which store the value of distances. DM stores distances in metres and centimeters and DB in feet and inches. Write a program that can read values for the class objects and add one object of DM with another object of DB.
Use a friend function to carry out the addition operation. The object that stores the results may be a DM object or DB object, depending on the units in which the results are required.The display should be in the format of feet and inches or metres and cenitmetres depending on the object on display.

6. Create a class rational which represents a numerical value by two double values- NUMERATOR & DENOMINATOR. Include the following public member Functions:
   a. constructor with no arguments (default).
   b. constructor with two arguments.
   c. void reduce( ) that reduces the rational number by eliminating the highest common factor between the numerator and denominator.
   d. Overload + operator to add two rational number.
   e. Overload >> operator to enable input through cin.
   f. Overload << operator to enable output through cout.

   Write a main ( ) to test all the functions in the class.

7. Consider the following class definition

```
class father {
        protected : int age;
public;
        father (int x) {age = x;}
        virtual void iam ( )
        { cout < < "I AM THE FATHER, my age is : "<< age<< end1:}
```

   Derive the two classes son and daughter from the above class and for each, define iam ( ) to write our similar but appropriate messages. You should also define suitable constructors for these classes. Now, write a main ( ) that creates objects of the three classes and then calls iam ( ) for them. Declare pointer to father. Successively, assign addresses of objects of the two derived classes to this pointer and in each case, call iam ( ) through the pointer to demonstrate polymorphism in action.

8. Write a program that creates a binary file by reading the data for the students from the terminal. The data of each student consist of roll no., name (a string of 30 or lesser no. of characters) and marks.

9. A hospital wants to create a database regarding its indoor patients. The information to store include
   a) Name of the patient
   b) Date of admission
   c) Disease
   d) Date of discharge

   Create a structure to store the date (year, month and date as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the to store the age of the patients. List the information about all the pediatric patients (less than twelve years in age).

10. Make a class **Employee** with a name and salary. Make a class **Manager** inherit from **Employee**. Add an instance variable, named department, of type string. Supply a method to **toString** that prints the manager's name, department and salary. Make a class **Executive** inherit from **Manager**. Supply a method **to String** that prints the string **"Executive"** followed by the information stored in the **Manager** superclass object. Supply a test program that tests these classes and methods.

11. Imagine a tollbooth with a class called toll Booth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both these to 0. A member function called payingCar ( ) increments the car total and adds 0.50 to the cash total. Another function, called nopayCar ( ), increments the car total but adds nothing to the cash total. Finally, a member function called displays the two totals. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC kay should cause the program to print out the total cars and total cash and then exit.

12. Write a function called reversit ( ) that reverses a string (an array of char). Use a for loop that swaps the first and last characters, then the second and next to last characters and so on. The string should be passed to reversit ( ) as an argument. Write a program to exercise reversit ( ). The program should get a string from the user, call reversit ( ), and print out the result. Use an input method that allows embedded blanks. Test the program with Napoleon's famous phrase, "Able was I ere I saw Elba)"

13. Create a base class called shape. Use this class to store two double type values that could be used to compute the area of figures. Derive two specific classes called triangle and rectangle from the base shape. Add to the base class, a member function get_data ( ) to initialize base class data members and another member function display_area ( ) to compute and display the area of figures. Make display_area ( ) as a virtual function and redefine this function in the derived classes to suit their requirements. Using these three classes, design a program that will accept dimensions of a triangle or a rectangle interactively and display the area. Remember the two values given as input will be treated as lengths of two sides in the case of rectangles and as base and height in the case of triangles and used as follows:
Area of rectangle = x * y
Area of triangle = ½ * x * y

4th Semester (CSE)

SOFTWARE ENGG. LAB
CSE-24P2

P **On Semester Evaluation: 120**
2 **End Semester Evaluation: 80**

NOTE: At Least 8 Experiments are to be performed from list below.

1. (i). Assume that you have been contracted by the university to develop an on-line university admission system(OLUAS). First act as a customer and specify the characteristics of a good system.
   (ii). Draw WBS for the OLUAS.

2. (i). Calculate effort and duration estimate for OLUAS.
   (ii). Draw timeline chart using some scheduling tool.

3. In OLUAS draw the following diagrams:
   (i). Context diagram
   (ii). Draw 2 level DFD's

4. (i). Write the use-case specification for Maintain Personal Planner describing the flow of events (basic flow and alternate flow), special requirements, pre-conditions and post-conditions, extension points if any for the use-case using Rational Rose.
   (ii). Add the use-case diagram to the browser clearly depicting the relationship among the use-cases.

5. (i). Create an Activity diagram that show the workflow for the use-case Maintain Personal Planner's basic flow of events.
   (ii). Add start and end nodes, State transitions, Decisions and Guard conditions to the activity diagram.

6. Identify the analysis classes for the Maintain Personal Planner (use-case) and identify the responsibilities of each class. Add classes under the analysis model in the browser and define the responsibilities for each class and select the appropriate stereotype for each class.

7. Create the sequence diagram for the above-mentioned Maintain Personal Planner use-case. Add object messages and responsibilities to object messages.

8. Create the class diagram showing the classes and relationship among classes that participate in Maintain Personal Planner's basic flow of events. Add association role names and multiplicity to the class diagram.

9. Create Activex .dll and Activex .exe diagram and do the Forward Engineering i.e generate the code from the diagram.

Books:

1. W. S. Jawadekar, Software Engineering Principle and Approaches. TMH. 2004.
2. Pressman S.Roger. Software Engineering. Tata McGraw-Hill.

3. Jalote Pankaj. An integrated approach to software engineering. Narosa Publishing House.
4. Sommerville Lan. Software Engineering. 5th ed. Addison Wesley-2000.
5. Fairley Richard, Software. Software Engineering Concepts. Tala McGraw-Hill.
6. Rational Rose Manual by IBM.

4th Semester (CSE)
MICROPROCESSORS & INTERFACING LAB
ECE-35P1

P **On Semester Evaluation: 120 Marks**
2 **End Semester Evaluation: 80 Marks**

LIST OF EXPERIMENTS:

Note: At least 15 experiments are to be performed from list below.

1. Familiarization with 8085 Trainer Kit by writing program for addition and subtraction of two numbers.
2. Write a program using 8085 for multiplication of two 8-bit numbers.
3. Write a program using 8085 to arrange block of data in descending order.
4. Write a program using 8085 to generate fibonacci series.
5. Write a program using 8085 to find out the smallest number in a string.
6. a) Familiarization with 8086 Trainer Kit.
   b) Familiarization with Digital I/O, ADC and DAC Cards.
   c) Familiarization with Turbo Assembler and Debugger S/Ws.
7. Write a program using 8086 to arrange block of data in ascending order.
8. Write a program using 8086 to find out any power of a number such that $Z=X^N$, where N is programmable and X is an unsigned number.
9. Write a program using 8086 to move a block of data from one location to another.
10. Write a program using 8086 to generate fibonacci series.
11. Write a program using 8086 to find out the largest number in a string.
12. Write a programmable delay routine to cause a minimum given delay and a maximum delay in the increments of small given increments.
13. Write a program using 8086 to find out even and odd numbers in a string.
14. WAP to implement keyboard interfacing with 8086 microprocessor using 8255 PPI Chip.
15. WAP to implement ADC 0808 interfacing with 8086 microprocessor using 8255 Chip.
16 WAP to interface 16-bit 8255 ports with 8086 microprocessor using the address of port A is FOH.
17. WAP for interfacing an I/P port 74LS245 to read the status of switches sw1-sw8. The switches when shorted, IP is 1 else it is 0. Store the status in BL register. The address of port is 740H.
18. Show 8259 interfacing connections with 8086 at the address 0740H.

4th Semester (CSE)
OPERATING SYSTEM LAB
CSE-24P3

P	**On Semester Evaluation: 120 Marks**
3	**End Semester Evaluation: 80 Marks**

1. Hands on the basic Linux Commands.
2. Implementation of contiguous, linked and indirect allocation strategies assuming randomly generated free space list.
3. Implementation of worst, best first fit for contiguous allocation assuming randomly generated free space list.
4. Implementation of compaction for the continually changing memory layout and calculate total movement of data.
5. Implementation of various CPU scheduling algorithms.
6. Calculation of external and internal fragmentation for different program and for different page size.
7. Implementation of Resource Allocation Graph.
8. Implementation of Banker's Algorithm.
9. Conversion of resource allocation graph to wait for graph.
10. Implementation of Producer –Consumer problem.
11. Implementation of Semaphore for concurrency.