II Year – I SEMESTER	T	P	C
	3+1	0	3

MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS

Unit – I: (*The Learning objective of this Unit is to understand the concept and nature of Managerial Economic s and its relationship with other disciplines, Concept of Demand and Demand forecasting)
Introduction to Managerial Economics and demand Analysis:
Definition of Managerial Economics and Scope-Managerial Economics and its relation with other subjects-Concepts of Demand-Types-Determents-Law of Demand its Exception-Elasticity of Demand-Types and Measurement-Demand forecasting and its Methods.
(**The Learner is equipped with the knowledge of estimating the Demand for a product and the relationship between Price and Demand)

Unit – II: (*The Learning objective of this Unit is to understand the concept of Production function, Input Output relationship, different Cost Concepts and Concept of Cost-Volume-Profit Analysis)
Production and Cost Analyses:
Production function-Isoquants and Isocosts-Law of Variable proportions-Cobb-Douglas Production function-Economics of Sale-Cost Concepts-Opportunity Cost-Fixed vs Variable Costs-Explicit Costs vs Implicit Costs-Out of Pocket Costs vs Imputed Costs-Cost Volume Profit analysis-Determination of Break-Even Point (Simple Problem)
(**One should understand the Cost Concepts for decision making and to estimate the least cost combination of inputs).

Unit – III: (*The Learning Objective of this Unit is t understand the Nature of Competition, Characteristics of Pricing in the different market structure and significance of various pricing methods)
Introduction to Markets, Theories of the Firm & Pricing Policies:
Market Structures: Perfect Competition, Monopoly and Monopolistic and Oligopoly – Features – Price, Output Determination – Managerial Theories of firm: Maris and Williamson’s models – Methods of Pricing: Limit Pricing, Market Skimming Pricing, Internet Pricing: Flat Rate Pricing, Usage sensitive, Transaction based pricing, Priority Pricing.
(** One has to understand the nature of different markets and Price Output determination under various market conditions)

Unit – IV: (*The Learning objective of this Unit is to know the different forms of Business organization and their Merits and Demerits both public & private Enterprises and the concepts of Business Cycles)
Types of Business Organization and Business Cycles:
Features and Evaluation of Sole Trader – Partnership – Joint Stock Company – State/Public Enterprises and their forms – Business Cycles – Meaning and Features – Phases of Business Cycle.
(**One should equipped with the knowledge of different Business Units)

Unit – V: (*The Learning objective of this Unit is to understand the different Accounting Systems preparation of Financial Statements and uses of different tools for performance evaluation)
Introduction to Accounting & Financing Analysis:
Introduction to Double Entry Systems – Preparation of Financial Statements-Analysis and Interpretation of Financial Statements-Ratio Analysis – Preparation of Funds flow cash flow statements (Simple Problems)
(**The Learner is able to prepare Financial Statements and the usage of various Accounting tools for Analysis)

Unit – VI: (*The Learning objective of this Unit is to understand the concept of Capital, Capitalization, Capital Budgeting and to know the techniques used to evaluate Capital Budgeting proposals by using different methods)
Capital and Capital Budgeting: Capital Budgeting: Meaning of Capital-Capitalization-Meaning of Capital Budgeting-Need for Capital Budgeting-Techniques of Capital Budgeting-Traditional and Modern Methods.

(**The Learner is able to evaluate various investment project proposals with the help of capital budgeting techniques for decision making)

Note: *Learning Objective
** Learning Assessment

TEXT BOOKS

1. Dr. N. Appa Rao, Dr. P. Vijay Kumar: ‘Managerial Economics and Financial Analysis’, Cengage Publications, New Delhi – 2011
2. Dr. A. R. Aryasri – Managerial Economics and Financial Analysis, TMH 2011
3. Prof. J.V.Prabhakara rao, Prof. P. Venkatarao. ‘Managerial Economics and Financial Analysis’, Ravindra Publication.

REFERENCES:

1. V. Maheswari: Managerial Economics, Sultan Chand.
2. Suma Damodaran: Managerial Economics, Oxford 2011.
3. Dr. B. Kuberudu and Dr. T. V. Ramana: Managerial Economics & Financial Analysis, Himalaya Publishing House 2011.
4. Vanitha Agarwal: Managerial Economics, Pearson Publications 2011.
5. Sanjay Dhameja: Financial Accounting for Managers, Pearson.
6. Maheswari: Financial Accounting, Vikas Publications.
7. S. A. Siddiqui & A. S. Siddiqui: Managerial Economics and Financial Analysis, New Age International Publishers, 2012

II Year – I SEMESTER	T	P	C
	3+1	0	3

OBJECT-ORIENTED PROGRAMMING THROUGH C++

Objectives: Expertise in object oriented principles and their implementation in C++

UNIT I :

Objectives: Exposure to basics of object oriented mode, C++ programming and I/O in C++

INTRODUCTION: Differences Between C And C++, The Object Oriented Technology , Disadvantage of Conventional Programming, Concepts of Object Oriented Programming, Advantages of OOP Structure of A C++ Program, Header Files And Libraries

INPUT AND OUTPUT IN C++ :

Introduction, Streams In C++ And Stream Classes, Pre-Defined Streams, Stream Classes, Formatted And Unformatted Data,Unformatted Console I/O Operations, Member Functions Of Istream Class, Formatted Console I/O Operations, Bit Fields, Flags Without Bit Field, Manipulators, User Defined Manipulators

UNIT II :

Objectives: Focus on Basic concept in C++ programming, Operators, control structures , functions, overloading, recursion

Tokens In C++, Variable Declaration And Initialization, Data Types, Operators In C And C++, Scope Access Operator, Namespace, Memory Management Operators, Comma Operator, Revision Of Decision Statements, Control Loop Statements

FUNCTIONS IN C++ : Introduction, Structure Of Function, Passing Arguments, Lvalues And Rvalues, Retrun By Reference, Returning More Values By Reference, Default Arguments, Const Arguments, Inputting Default Arguments, Inline Functions, Function Overloading, Principles Of Function Overloading, Recursion

UNIT III :

Objectives: Acquaintance with classes, objects and member functions

CLASSES AND OBJECTS : Introduction, Classes In C++, Declaring Objects, Access Specifiers And Their Scope, Member Functions, Outside Member Function As Inline, Data Hiding or Encapsulation, Classes, Objects and Memory, Static Member Variables, Static Member Functions Static Object, Array Of Objects, Objects As Function Arguments, Friend Functions, The Const Member Functions, The Volatile Member Function, Recursive Member Function, Local Classes, Empty, Static And Const Classes, Member Function and Non- Member Function, Overloading Member Functions, Nested Class

UNIT IV :

Objectives: Focus on constructors , destructors, variants in them, operator overloading, type conversions

CONSTRUCTORS AND DESTRUCTORS : Introduction, Characteristic Of Constructors & Destructors, Applications With Constructors, Parameterized Constructor, Overloading Constructors (Multiple Constructors), Array Of Objects Using Constructors, Constructors With Default Arguments, Copy Constructors, The Const Objects, Destructors, Calling Constructors And Destructors, Qualifier And Nested Classes, Anonymous Objects, Private Constructors And Destructors, Dynamic Initialization Using Constructors, Dynamic Operators and Constructors, Recursive Constructor, Constructor and Destructor With Static Members, Local Vs. Global Object

OPERATOR OVERLOADING AND TYPE CONVERSION : Introduction, Overloading Unary Operators, Constraint on Increment And Decrement Operators, Overloading Binary Operators, Overloading With Friend Function, Overloading Assignment Operator (=), Type Conversion, Rules For Overloading Operators, One Argument Constructor And Operator Function, Overloading Stream Operators

UNIT V :

Objective: Concentration on inheritance, types of inheritance, polymorphism, virtual functions

INHERITANCE : Introduction, Reusability, Access Specifiers And Simple Inheritance, Protected Data With Private Inheritance, Types Of Inheritances(Single Inheritance, Multilevel Inheritance, Multiple Inheritance, Hierarchical Inheritance, Hybrid Inheritance, Multipath Inheritance), Virtual Base Classes, Constructors, Destructors, And Inheritance, Object As A Class Member, Abstract Classes, Qualifier Classes And Inheritance, Constructor In Derived Class, Pointers And Inheritance, Overloading Member Function, Advantages Of Inheritance, Disadvantages Of Inheritance.

BINDING, POLYMORPHISM AND VIRTUAL FUNCTIONS: Introduction, Binding In C++, Static (Early) Binding, Dynamic (Late) Binding, Pointer To Base And Derived Class Objects, Virtual Functions, Rules For Virtual Functions, Array Of Pointers, Pure Virtual Functions, Abstract Classes, Working Of Virtual Functions, Virtual Functions In Derived Classes, Object Slicing, Constructors And Virtual Functions, Virtual Destructors, Destructor And Virtual Functions.

UNIT VI :

Objectives: Focus on Files, File operations, generic programming, templates, function templates, Exception handling

APPLICATIONS WITH FILES: Introduction, File Stream Classes, File Opening Modes, File Pointers And Manipulators, Manipulators With Arguments, Sequential Access Files, Binary And ASCII Filesrandom Access Operation,

GENERIC PROGRAMMING WITH TEMPLATES : Introduction, Need Of Template, Definition Of Class Template, Normal Function Template, Working Of Function Templates, Class Template With More Parameters, Functions Templates With More Arguments, Overloading Of Template Functions, Member Function Templates,Recursion With Template Function, Class Template With Overloaded Operators, Class Template Revisited, Class Templates And Inheritance, Container Classes , Types Of Containers, Container Adaptors, Iterators

EXCEPTION HANDLING : Introduction, Principles Of Exception Handling,The Keywords Try, Throw And Catch , Exception Handling Mechanism, Multiple Catch Statements, Catching Multiple Exceptions, Re-Throwing Exception, Specifying Exception, Exceptions In Constructor And Destructors, Controlling Uncaught Exceptions, Class Template With Exception Handling

TEXT BOOKS :

1. Programming In C++ , Ashok N Kamthane. Pearson 2nd Edition.
2. Object Oriented Programming C++ , Joyce Farrell, Cengage
3. Mastering C ++, Venugopal, Rajkumar, Ravi kumar TMH
4. Object Oriented Programming with C++, 2nd ed, Sourav Sahay, OXFORD

REFERENCE BOOKS:

1. The Complete Reference, C++, 4ed, Herbert Schildt, TMH

II Year – I SEMESTER	T	P	C
	3+1	0	3

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

Objectives: Acquaintance with the basic mathematical implication for computer science, applications of mathematics in computer science

UNIT I:

Objective: Acquiring the relevance of statements, inferences and predicates in computer science

Mathematical Logic :

Propositional Calculus: Statements and Notations, Connectives, Truth Tables, Tautologies, Equivalence of Formulas, Duality law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, **Consistency of Premises, Indirect Method of Proof.**

Predicate calculus: Predicative Logic, Statement Functions, Variables and Quantifiers, Free & Bound Variables, Inference theory for predicate calculus.

UNIT II :

Objective: Overview of number theory, basic algorithms in number theory and mathematical induction

Number Theory & Induction:

Properties of integers, Division Theorem, The Greatest Common Divisor, Euclidean Algorithm, Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic (Fermat's Theorem and Euler 's Theorem)

Mathematical Induction: Principle of Mathematical Induction, exercises

UNIT III:

Objective: Focuses on sets and relations and their operations, relations and functions

Set Theory:

Introduction, Operations on Binary Sets, Principle of Inclusion and Exclusion

Relations: Properties of Binary Relations, Relation Matrix and Digraph, Operations on Relations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering Relations, Hasse Diagrams.

Functions: Bijective Functions, Composition of Functions, Inverse Functions, Permutation Functions, Recursive Functions

UNIT IV:

Objectives: Exposure of graphs, their representation, types, trees and tree variants

Graph Theory:

Basic Concepts of Graphs, Sub graphs, Matrix Representation of Graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multigraphs, (Problems and Theorems without proofs)

Planar Graphs, Euler's Formula, Graph Colouring and Covering, Chromatic Number,(Problems and Theorems without proofs)

Trees, Directed trees, Binary Trees, Decision Trees,

Spanning Trees: Properties, Algorithms for Spanning trees and Minimum Spanning Tree.

UNIT V:

Objective: Overview of algebraic structures, Group theory, Binomial theorem, permutations and combinations

Algebraic Structures: Lattice: Properties, Lattices as Algebraic Systems, Algebraic Systems with one Binary Operation, Properties of Binary operations, Semi groups and Monoids: Homomorphism of Semi groups and Monoids, Groups: Abelian Group, Cosets, Subgroups (Definitions and Examples of all Structures) Algebraic Systems with two Binary Operations: Rings

Combinatorics: Basic of Counting, Permutations, Derangements, Permutations with Repetition of Objects, Circular Permutations, Restricted Permutations, Combinations, Restricted Combinations, Pigeonhole Principle and its Application.

Binomial Theorem: Binomial and Multinomial Coefficients, Generating Functions of Permutations and Combinations, The Principles of Inclusion – Exclusion.

UNIT VI:

Objective: Overview of generating functions, recurrence relations and solving recurrence relations

Recurrence Relation:

Generating Function of Sequences, Partial Fractions, Calculating Coefficient of Generating Functions Recurrence Relations, Formulation as Recurrence Relations, Solving linear homogeneous recurrence Relations by substitution, generating functions and The Method of Characteristic Roots.

Solving Inhomogeneous Recurrence Relations

TEXT BOOKS :

1. Discrete Mathematical Structures with Applications to Computer Science, Tremblay, Manohar, TMH
2. Discrete Mathematics for Computer Scientists & Mathematicians, 2/e, Mott, Kandel, Baker, PHI
3. Discrete Mathematics, Swapan Kumar chakrborthy, Bikash kanti sarkar, OXFORD
4. Discrete Mathematics and its Applications with combinatorics and graph theory, 7th ed, Rosen, TMH
5. Discrete Mathematics, Theory and Applications, Malik sen, Cengage
6. Discrete mathematics and Graph theory, 3rd ed, Biswal, PHI

REFERENCE BOOKS:

1. Discrete Mathematics, Proofs, Structures and applications, 3rd ed, CRC Press
2. Discrete Mathematics, S.Santha, Cengage
3. Discrete Mathematics with Applications, Thomas Koshy, Elsevier
4. Discrete Mathematics,2/e, JK Sharma ,Macmillan

II Year – I SEMESTER

T	P	C
3+1	0	3

DIGITAL LOGIC DESIGN

UNIT I: Number Systems

Binary, Octal, Decimal, Hexadecimal Number Systems. Conversion of Numbers From One Radix To Another Radix, r's Complement and (r-1)'s Complement Subtraction of Unsigned Numbers, Problems, Signed Binary Numbers, Weighted and Non weighted codes

UNIT II:Logic Gates And Boolean Algebra

Basic Gates NOT, AND, OR, Boolean Theorms,Complement And Dual of Logical Expressions, Universal Gates, Ex-Or and Ex-Nor Gates, SOP,POS, Minimizations of Logic Functions Using Boolean Theorems, Two level Realization of Logic Functions Using Universal Gates

Gate Level Minimization: Karnaugh Map Method (K-Map): Minimization of Boolean Functions maximum upto Four Variables, POS and SOP, Simplifications With Don't Care Conditions Using K-Map.

UNIT III: Combinational Logic Circuits

Design of Half Adder, Full Adder, Half Subtractor, Full Subtractor, Ripple Adders and Subtractors, Ripple Adder/Subtractor Using Ones and Twos Complement Method. Design of Decoders, Encoders, Multiplexers, Demultiplexers, Higher Order Demultiplexers and Multiplexers, Priority Encoder, Code Converters, Magnitude Comparator.

UNIT IV: Introduction to Sequential Logic Circuits

Classification of Sequential Circuits, Basic Sequential Logic Circuits: Latch and Flip-Flop, RS- Latch Using NAND and NOR Gates, Truth Tables. RS,JK,T and D Flip Flops , Truth and Excitation Tables, Conversion of Flip Flops. Flip Flops With Asynchronous Inputs (Preset and Clear).

UNIT V: Registers and Counters

Design of Registers, Buffer Register, Control Buffer Registers, Bidirectional Shift Registers, Universal Shift Register, Design of Ripple Counters, Synchronous Counters and Variable Modulus Counters, Ring Counter, Johnson Counter.

UNIT VI: Introduction to Programmable Logic Devices (PLOs)

PLA, PAL, PROM. Realization of Switching Functions Using PROM, PAL and PLA. Comparison of PLA, PAL and PROM.

TEXT BOOKS :

1. Digital Design ,4/e, M.Morris Mano, Michael D Ciletti, PEA
2. Fundamentals of Logic Design, 5/e, Roth, Cengage

REFERENCE BOOKS

1. Switching and Finite Automata Theory,3/e,Kohavi, Jha, Cambridge.
2. Digital Logic Design, Leach, Malvino, Saha,TMH
3.Modern Digital Electronics, R.P. Jain, TMH

II Year – I SEMESTER	T	P	C
	3+1	0	3

DATA STRUCTURES

Objectives: Comprehensive knowledge of data structures and ability to implement the same in software applications

UNIT I:

Objective: exposure to algorithmic complexities, recursive algorithms, searching and sorting techniques

Preliminaries of algorithm, Algorithm analysis and complexity,

Data structure- Definition, types of data structures

Recursion: Definition, Design Methodology and Implementation of recursive algorithms, Linear and binary recursion, recursive algorithms for factorial function, GCD computation, Fibonacci sequence, Towers of Hanoi, Tail recursion

List Searches using Linear Search, Binary Search, Fibonacci Search

Sorting Techniques: Basic concepts, Sorting by : insertion (Insertion sort), selection (heap sort), exchange (bubble sort, quick sort), distribution (radix sort) and merging□(merge sort)□Algorithms.

UNIT II:

Objectives: Applying stack and queue techniques for logical operations

Stacks and Queues: Basic Stack Operations, Representation of a Stack using Arrays, Stack Applications: Reversing list, Factorial Calculation, Infix to postfix Transformation, Evaluating Arithmetic Expressions.

Queues: □Basic Queues Operations, Representation □of a □Queue using array, Implementation of Queue Operations using Stack,□ Applications of Queues-Round robin Algorithm,□Circular Queues, □ Priority Queues.

UNIT III:

Objectives: Exposure to list representation models in various types of applications

Linked Lists: Introduction, single linked list, representation of a linked list in memory, Operations on a single linked list, Reversing a single linked list, applications of single linked list to represent polynomial expressions and sparse matrix manipulation, Advantages and disadvantages of single linked list, Circular linked list, Double linked list

UNIT IV:

Objectives: Implementation of tree implementation in various forms

Trees: Basic tree concepts,□Binary Trees: Properties, Representation of Binary Trees using arrays and linked lists, operations on a Binary tree , Binary Tree Traversals (recursive), Creation of binary tree from in, pre and post order traversals

UNIT-V:

Objectives: Advanced understanding of other variants of trees and their operations

Advanced concepts of Trees: Tree Travels using stack (non recursive), Threaded Binary Trees. Binary search tree, Basic concepts, BST operations: insertion, deletion, Balanced binary trees – need, basics and applications in computer science (No operations)

UNIT VI:

Objectives: orientation on graphs, representation of graphs, graph traversals, spanning trees

Graphs: Basic concepts, Representations of Graphs: using Linked list and adjacency matrix, Graph algorithms

Graph Traversals (BFS & DFS), applications: Dijkstra's shortest path, Transitive closure, Minimum Spanning Tree using Prim's Algorithm, warshall's Algorithm(**Algorithemic Concepts Only, No Programs required**).

TEXT BOOKS:

1. Data Structure with C, Seymour Lipschutz, TMH
2. Data Structures using C,Reema Thareja, Oxford
3. Data Structures, 2/e, Richard F, Gilberg , Forouzan, Cengage
4. Data structures and algorithm analysis in C, 2nd ed, mark allen weiss

REFERENCE BOOKS:

1. Data Structures and Algorithms, 2008,G.A.V.Pai, TMH
2. Classic Data Structures, 2/e, Debasis ,Samanta,PHI,2009
3. Fundamentals of Data Structure in C, 2/e, Horowitz,Sahni, Anderson Freed,University Prees

II Year – I SEMESTER

	T	P	C
	0	3	2

OBJECT-ORIENTED PROGRAMMING LAB

1. Write a C++ program illustrating Variable Scope.
2. Write a C++ program illustrating Swap integer values by reference.
3. Write a C++ program illustrating Checking whether the number is even or odd using Ternary operator.
4. Write a C++ program illustrating a program to find the roots of a quadratic equation .Use switch statements to handle different values of the discriminant (b^2-4*a*c).
5. Write a C++ program illustrating interactive program to multiply 2 variables after checking the compatibility.
6. Write a C++ program illustrating interactive program for computing the roots of a quadratic equation by handling all possible cases. Use streams to perform I/O operations.
7. Write a C++ program illustrating to sort integer numbers.
8. Write a C++ program illustrating factorial using recursion.
9. Write a C++ program illustrating pass by value, pass by reference, pass by address.
10. Write a C++ program illustrating Function overloading.
11. Write a C++ program illustrating an interactive program for swapping integer, real, and character type variables without using function overloading .Write the same program by using function overloading features and compare the same with its C counterpart.
12. Write a C++ program illustrating inline functions.
13. Write a C++ program illustrating Friend function.
14. Write a C++ program illustrating Exception handling.
15. Write a C++ program illustrating Function template.
16. Write a C++ program illustrating Overloading increment, decrement, binary+&<< operator.
17. Write a C++ program illustrating Virtual function.
18. Write a C++ program illustrating an interactive program to process complex numbers .It has to Perform addition, subtraction, multiplication, and division of complex numbers. print results in x+iy form. Create a class for the complex number representation.
19. Write a C++ program illustrating user defined string processing functions using pointers (string length, string copy, string concatenation)
20. Write a C++ program illustrating Constructor overloading (Both parameterised and default).
21. Write a C++ program illustrating Copy constructor.
22. Write a C++ program illustrating access data members & member functions using ‘THIS’ pointer.
23. Write a C++ program illustrating for overloading ++ operator to increment data.
24. Write a C++ program illustrating overloading of new and delete operator.
25. Write a C++ program illustrating Abstract classes.
26. Write a C++ program illustrating Inheritance (Multiple, Multilevel, Hybrid).
27. Write a C++ program illustrating Virtual classes & virtual functions.
28. Write a C++ program illustrating overloading function template.
29. Write a C++ program illustrating Class template.

II Year – I SEMESTER

T	P	C
0	3	2

DATA STRUCTURES LAB

Exercise 1:

Write recursive program which computes the n^{th} Fibonacci number, for appropriate values of n.
Analyze behavior of the program Obtain the frequency count of the statement for various values of n.

Exercise 2:

Write recursive program for the following

a) Write recursive and non recursive C program for calculation of Factorial of an integer

b) Write recursive and non recursive C program for calculation of GCD (n, m)

c) Write recursive and non recursive C program for Towers of Hanoi : N disks are to be transferred from peg S to peg D with Peg I as the intermediate peg.

Exercise 3:

a) Write C program that use both recursive and non recursive functions to perform Linear search for a Key value in a given list.

b) Write C program that use both recursive and non recursive functions to perform Binary search for a Key value in a given list.

c) Write C program that use both recursive and non recursive functions to perform Fibonacci search for a Key value in a given list.

Exercise 4:

a) Write C program that implement Bubble sort, to sort a given list of integers in ascending order

b) Write C program that implement Quick sort, to sort a given list of integers in ascending order

c) Write C program that implement Insertion sort,to sort a given list of integers in ascending order

Exercise 5:

a) Write C program that implement heap sort, to sort a given list of integers in ascending order

b) Write C program that implement radix sort, to sort a given list of integers in ascending order

c) Write C program that implement merge sort, to sort a given list of integers in ascending order

Exercise 6:

a) Write C program that implement stack (its operations) using arrays

b) Write C program that implement stack (its operations) using Linked list

Exercise 7:

a) Write a C program that uses Stack operations to Convert infix expression into postfix expression

a) Write C program that implement Queue (its operations) using arrays.

b) Write C program that implement Queue (its operations) using linked lists

Exercise 8:

a) Write a C program that uses functions to create a singly linked list

b) Write a C program that uses functions to perform insertion operation on a singly linked list

c) Write a C program that uses functions to perform deletion operation on a singly linked list

Exercise 9:

a) Adding two large integers which are represented in linked list fashion.

b) Write a C program to reverse elements of a single linked list.

c) Write a C program to store a polynomial expression in memory using linked list

d) Write a C program to representation the given Sparse matrix using arrays.

e) Write a C program to representation the given Sparse matrix using linked list

Exercise10:

a) Write a C program to Create a Binary Tree of integers

b) Write a recursive C program for Traversing a binary tree in preorder, inorder and postorder.

c) Write a non recursive C program for Traversing a binary tree in preorder, inorder and postorder.

d) Program to check balance property of a tree.

Exercise 11:

a) Write a C program to Create a BST

b) Write a C program to insert a node into a BST.

c) Write a C program to delete a node from a BST.

II Year – I SEMESTER

T	P	C
0	3	2

DIGITAL LOGIC DESIGN LAB

List of Experiments:

1) Verification of Basic Logic Gates.
2) Implementing all individual gates with Universal Gates NAND & NOR.
3) Design a circuit for the given Canonical form, draw the circuit diagram and verify the De-Morgan laws.
4) Design a Combinational Logic circuit for 4x1 MUX and verify the truth table.
5) Design a Combinational Logic circuit for 1x4 De- MUX and verify the truth table.
6) Verify the ***data read*** and ***data write*** operations for the IC 74189.
7) Design a Gray code encoder and interface it to SRAM IC 74189 for write operation display on 7-segment.
8) Design a Gray code De-coder and interface it to SRAM IC 74189 for read operation display it on 7-segment.
9) Construct Half Adder and Full Adder using Half Adder and verify the truth table.
10) Verification of truth tables of the basic Flip- Flops with ***Synchronous*** and ***Asynchronous*** modes.
11) Implementation of Master Slave Flip-Flop with J-K Flip- Flop and verify the truth table for ***race around*** condition**.**
12) Design a Decade Counter and verify the truth table.
13) Design the Mod 6 counter using D-Flip -Flop.
14) Construct 4-bit ring counter with T-Flip –Flop and verify the truth table.
15) Design a 8 – bit right Shift Register using D-Flip -Flop and verify the truth table.

II Year – I SEMESTER

T	P	C
0	0	1

Seminar