

BCA Subjects	Semester	Credits
Discrete Mathematics	1	4
Professional Communication Skills 1	1	4
Fundamentals of Computers	1	4
C-Programming	1	4
Lab - C Programming	1	2
Professional Communication Skills II	2	4
Introduction to Database Management System	2	4
Introduction to Operating Systems	2	4
Object Oriented Programming using C++	2	4
Lab - Object Oriented Programming using C++	2	2

## Semester 1

### Subject: Discrete Mathematics

CO#	Course Outcome
CO1	Understand the Propositional Logic and Set Theory
CO2	Understand the Recurrence Relations and Graph
CO3	Understand the Tree and Directed Graphs
CO4	Understand the Matrices and Vector Spaces of Graphs and Graph Algorithms

### Chapter 1: Logic

1.1 Propositional Logic - 1.2 Predicates and Quantifiers - 1.3 Rules of Inference: Argument in propositional Logic

### Chapter 2: Set Theory

2.1 Basic Set Theory - 2.1.1 Union and Intersection of Sets - 2.1.2 Set Difference, Set Complement and the Power Set - 2.2 Relations and Functions - 2.2.1 Composition of Functions - 2.2.2 Equivalence Relation - 2.3 Advanced topics in Set Theory and Relations - 2.3.1 Families of Sets - 2.3.2 More on Relations

### Chapter 3: Recurrence Relations

3.1 Recurrence Relations: Introduction, Formation. - 3.2 Linear Recurrence Relations with constant coefficients. - 3.3 Homogeneous Solutions. - 3.4 Particular Solutions.

### Chapter 4: Graph

4.1 Definitions - 4.2 Walks, Trails, Paths, Circuits, Connectivity, Components - - 4.3 Graph Operations - 4.4 Cuts - 4.5 Labeled Graphs and Isomorphism

### Chapter 5: Tree

5.1 Trees and Forests- 5.2 (Fundamental) Circuits and (Fundamental) Cut Set

### Chapter 6: Directed Graphs

6.1 Definition - 6.2 Directed Trees -6.3 Acyclic Directed Graphs

### Chapter 7: Matrices and Vector Spaces of Graphs

Matrix Representation of Graphs - Cut Matrix - Circuit Matrix - An Application: Stationary Linear Networks  
- Matrices over GF(2) and Vector Spaces of Graphs

### Chapter 8: Graph Algorithms

8.1 Computational Complexity of Algorithms - Reachability: Warshall's Algorithm - Depth-First and Breadth-First Searches - The Lightest Path: Dijkstra's Algorithm - The Lightest Path: Floyd's Algorithm - The Lightest Spanning Tree: Kruskal's and Prim's Algorithms - The Lightest Hamiltonian Circuit (Travelling Salesman's Problem): The Annealing Algorithm and the Karp–Held Heuristics - Maximum Matching in Bipartite Graphs: The Hungarian Algorithm - Maximum Flow in a Transport Network: The Ford–Fulkerson Algorithm

=====

### Subject: Professional Communication Skills - 1

CO#	Course Outcome
CO1	To know meaning and significance communication skills
CO2	To define process of communication and various forms of communication
CO3	To understand and apply effective listening skills and body language in communication
CO4	To inculcate significance of interpersonal communication and ways to enhance interpersonal relationships

### Chapter: 1 Communication Skills

Introduction - Definition of Communication - Significance - Disasters of Non-communication - Communication Gap: Gender Gap, Psychological Gap, Generation Gap, Spatial Gap, Cultural Gap, Knowledge Gap, Status Gap, Credibility Gap - Communication Skills - Summary - Self Assessment Questions

### Chapter: 2 Process of Communication

Introduction - Process of Communication: Action Model/ Bull's-eye Theory, Interaction Model/ Ping-Pong Theory - Feedback - Key for Effective Communication - Guidelines to effective communication - Forms of Communication: Communication Media, Oral Communication, Written (Print) Communication, Nonverbal Communication - Summary - Self Assessment Questions

### Chapter: 3 Listening Skills

Introduction - Concept of Listening - Significance of Listening - Types of Listening: Active Listening, Inactive Listening, Attentive Listening, Appreciative Listening, Empathetic Listening, Sympathetic Listening, Inattentive Listening - Listening Skills - Active and Attentive Listening: Processing Strategies, Bottom-up Processing, Top-down Processing, Barriers, Guides to Effective Listening - Benefits of Listening - Summary - Self Assessment Questions

### Chapter: 4 Body Language

Introduction - Concept of Body Language - Effects of Right Body Language - Postures and Meanings - Right Posture - its Importance - Summary - Self Assessment Questions

**Chapter 5: Interpersonal Communication**

Introduction: interpersonal communication - Communication and Emotion - Definition of Interpersonal Communication - Significance of Interpersonal Relationships and communication - Enhance your interpersonal communication and relationships - Self Assessment Question



**Subject: Fundamental of Computer**

CO#	Course Outcome
CO1	Understand the building blocks of a computer, Data Representation and Algebra
CO2	Understand the Combinational Circuits and Sequential Circuits
CO3	Understand the Memory System
CO4	Understand CPU Organization

**Chapter-1. Introduction to Digital Computer**

Structure-Introduction, Digital Computer, Layers in Modern Computer, Types of Software, Programming Languages Software

**Chapter-2. Data Representation and Boolean Algebra**

Structure-Introduction, Decimal Number System, Binary Number System, Octal Number System, Hexadecimal Number System, Signed Binary Numbers, Binary Arithmetic, Digital Codes, Logic gates, De Morgans's Theorems, Duality Theorem, Boolean Algebra Rules and Laws, Standard (canonical) Representations for Logical Functions (SOP AND POS), K-Map

**Chapter-3. Combinational Circuits and Sequential Circuits**

Structure-Introduction, Adders, Encoder, Decoder, Multiplexer, Demultiplexer, Sequential Circuits, Flip Flops, Shift Registers, Introduction to Counters

**Chapter-4. Memory System**

Structure-Introduction, Classification of Memory, Memory Hierachy, Primary Memory, ROM, Cache Memory, DMA (Direct Memory Access)

**Chapter-5. CPU Organization**

Structure-Introduction, CPU building blocks, CPU Registers and Bus Characteristics, Interface, Local Bus, Addressing Modes, Interrupts, Instruction and Execution cycle, Hardwired and Micro Program control, RISC and CISC, Pipelining: Data path, Time Space Diagram, Hazards



## Subject: C-Programming

CO#	Course Outcome
CO1	Understand the History of C and fundamentals required for developing programs
CO2	Understand the Looping structures and implement programs
CO3	Understand the Arrays, pointers, Functions, File Handling etc
CO4	Understand the advanced concepts in C

### Chapter 1. An Overview of C

History, Developing of C, Where C Stands?, Program Developing Cycle, The Form of a C Program, Structure of a 'C' Program, Compilers and Interpreters, Executing A 'C' Program

### Chapter 2. Variable, Data Types, Operator And Expression

Introduction, Character Set, C Tokens, Data Types in c, Variables, Data Declaration and Definitions, User Defined Type Declaration, Operation and Expressions, Type Conversation in Expressions, Precedence and Associativity of Operators

### Chapter 3. Built In I/O Functions

Introduction, Unformatted Console I/O Operations, Formatted Console I/O Operations

### Chapter 4. Control Statement

Introduction, Selection/Decision Making Statements, Iterative Statements, Jump Statements, Compound Statement, Null Statement

### Chapter 5. Array and String

Introduction, Array Declaration, One Dimensional Array, Multidimensional Arrays, Strings

### Chapter 6. Pointers

Introduction, Memory Organization, Basic of Pointers, Application of Pointers, Using pointers, Pointers Expression, Precedence of & AND \* Operators, Pointer to pointer, Pointers to Constant Objects, Constant Pointer, Dynamic Memory Allocation, Pointer and Arrays, Pointers and Character String, Array of Pointers

### Chapter 7. Function

Introduction, What is a Function?, Functions and Structured Programming, How a Function Works?, Library and User Defined Functions, Function Declaration and Definition, Writing a Function, Calling a Function, Types of Functions, Methods of passing Arguments, Arrays and Functions, Pointers and Function, Recursion

### Chapter 8. Storage Classes and Scope

Meaning of Terms, Scope, Storage Classes

## Chapter 9. Structures, Union, Enumeration and typedef

Structures, Structures and Enumerated Data Type, Union, Difference between Structure and Union

## Chapter 10. C Preprocessor

What is a Preprocessor?, Preprocess Directives

## Chapter 11. File Handling

Introduction, Stream, Types of Files, Operation on a File, Error Handling during I/O Operations, Random Access to Files

## Chapter 12. Bitwise Operators

Introduction, Application, Bit Fields

## Chapter 13. Graphic In C

Introduction, Basic Concepts, Drawing Simple Graphic Objects, Output Text

## Chapter 14. Command Line Arguments

Introduction, Advantages of Command Line Arguments

=====

**Subject : Lab - C-Programming**

CO#	Course Outcome
CO1	Understand the History of C and fundamentals required for developing programs
CO2	Understand the Looping structures and implement programs
CO3	Understand the Arrays, pointers, Functions, File Handling etc.
CO4	Understand the advanced concepts in C

1. Write a program to print multiplication tables up to 10.
2. Write a program to find sum of 'N' numbers using Recursion.
3. Write a program to find the sum of natural numbers using for loop.
4. Write a program to make a simple and basic calculator using switch case.
5. Write a program to store information of student and display it using Structure.
6. Write a program to find out the sum of series  $1 + 2 + \dots + n$ .
7. Write a program to swap variables value Using Call by Reference.

8. Write a program to check whether the entered number is Armstrong or not.
9. Write a program to sort the array in ascending order.
10. Write a program to fill colour in the circle.

=====

**Semester II**

**Professional Communication Skills - 2**

CO#	Course Outcome
CO1	To help students in analyzing various perspectives attitudes and steps to follow in order to develop positive attitude
CO2	To understand importance to time management and stress management; also to analyse how improper time management triggers stress at work place
CO3	To learn about meaning and application of emotional intelligence in day to day activities
CO4	To help students to understand meaning and practical applications of goals and goal setting

**Syllabus:**

**Chapter: 1 Attitudes**

Introduction : Concept - Types of Attitude: (Positive VS Negative Attitude, Winning VS Losing Attitude) - Formation of Attitude- Importance of Positive Attitude: Benefits of Individuals, Benefits of Organizations - Steps in developing positive Attitude: Building Cognitive Component, Building affective Component, Behavioral Component - Summary - Self Assessment Questions

**Chapter: 2 Stress Management**

Introduction - Definitions: Dynamic Condition, Desire, Opportunity, constraint or demand, Important but uncertain outcome - Types of Stress: High Stress, Moderate Stress, Low or Mild stress, Distress, Eustress - Linkage between Stress and Time Management - Stress levels and Consequences: General Consequences for the individuals, Consequences for the Organization, Consequences for the Family - Sources of Stress - Stress Coping Ability: Stress Threshold, Stress Resiliency - Measures to Manage Stress - Principles of Stress Management - Summary - Self Assessment Questions

**Chapter 3: Emotional Intelligence**

Introduction - Concept and Definitions - Elements of Emotional Intelligence - Organizational Application - Conflict Management - Summary - Self Assessment Questions

**Chapter 4: Goal Setting**

Introduction - Concept of Goals - Goals and Periodicity - Characteristics of Goals: Objectives are Futuristic, Objectives are Concrete, Objectives are Attainable, Objectives are Measurable, Objectives should be Acceptable, Guidelines to Personality Development - Importance of Goals - Significance of Goal setting: Goal setting is a basic function of Management, Goal setting replaces hunches by Judgment, Goal setting involves rational processes, Goal setting involves balancing - Activity in Goal Setting - Common Obstacles to Goal Achievement - Methods of Achieve set goals: Work Planning, Progress chasing, Performance Enhancing - Summary - Self Assessment Questions

## Chapter 5: Time Management

Introduction - Importance of Time: Survival of the Fastest, Chance of Recovery, Time impacts Health, Prerequisite for success - Techniques of Time Management - Prioritization of activities: The 80/20 Rule, Goals and Tasks Analysis, Set Goals, Identify tasks, Identifying your strength and weaknesses, Ways of Organizing Work, Scheduling, Weekly Activity descriptions and times - Avoiding Time waters: Process Related Factors, Procrastination, perfectionism, Lack of Self-discipline, Crisis Management, Interruptions - Summary - Self Assessment Questions.

=====

## Subject : Introduction to Database Management System

CO#	Course Outcome
CO1	Understand the fundamentals of DB, ER Model, Relational Model
CO2	Understand the Conventional Data Models, Relational DB, Storage and Files
CO3	Understand the Transaction Concurrency, Control, Crash Recovery and Backup
CO4	Understand the Security, Privacy and Non SQL DB

### Chapter -1 Basic Concept

Database and Database User, Data Independence

### Chapter-2 Database Design Using ER Model

Data Models, ER Model for Conceptual Design, Relationships, Relationship Sets, Mapping Cardinalities, Types of Keys, ER-Diagram

### Chapter-3 Relational Model

Relational Data Model, Relational Algebra, SQL-A Relational Database Language, Indexing, Views, Security in SQL, Triggers

### Chapter-4 Conventional Data Models and Systems

Networks Data Models and IDMS System, Hierarchical Data Model

### Chapter 5-Relational Database

Introduction , Function Dependencies , Undesirable Properties of A Bad Database Design , E.F Codd's Rules, Steps Followed By Application Developer, Normalization Process, Denormalisation, Loseless Joins, Decompositions

## Chapter 6-Storage and File Structure

Introduction, Overview of Physical Storage Media, Magnetic Disks, Raid, Tertiary Storage, Storage Access, Organization of Records in Files, Data Dictionary Storage, Factors Used for Evaluation of The Above Techniques

## Chapter 7-Transaction and Concurrency Control

Concept of Transaction, Properties of Transaction , State of Transaction Implementation of Atomicity and Durability, Concurrency Executions, Concurrency Control Techniques, Concurrency Control

## Chapter 8-Crash Recovery and Backup

Why Recover IS Needed?, Storage Structure, Recovery and Atomicity, Failure With Loss of Nonvolatile Storages, Recovery and Atomicity, Failure with Loss of Nonvolatile Storage, Recovery From Catastrophic Failure, Remote Backup Systems

## Chapter 9-Security and Privacy

Introduction, Discretionary Access Control Method , Mandatory Access Control Method, Uses of View in Security Enforcement , Overview Of Encryption Technique for database

## Chapter 10-NON-SQL Database

A Relational Database Management System, NoSQL Emerged From a Need, What is NoSQL

=====

## Student: Operating System

CO#	Course Outcome
CO1	Understand the basics of OS and Process Management
CO2	Understand the CPU Scheduling, Process Synchronization, Deadlock and Memory Management
CO3	Understand the File and Disk Management
CO4	Understand the Distributed OS

## Chapter 1-Introduction

Introduction, View of Operating System, View of Operating System, System Calls , System Programs, Operating System Structure, Concept of Virtual Machine

## Chapter 2-Process Management

Process Concept, Process Control Block, Process Scheduling, Process Operations, Inter process Communication, Communication in Client-Server, RTOS (Real Time Operating System)



**Chapter 3-CPU Scheduling**

Introduction, Scheduling Concept, Cpu-I/O Burst Cycle, Scheduling Criteria, Scheduling Algorithms, Scheduling Evaluation, Simulation

**Chapter 4-Process Synchronization and Deadlock**

Synchronization Concept, Critical Section Problem, Monitor, Deadlock Concepts, Deadlock Prevention and Avoidance, Deadlock Detection, Deadlock Recovery

**Chapter 5-Memory Management**

Concept, Memory Management Techniques, Contiguous and Non-Contiguous Allocation, Logical to Physical Address, Paging, Segment with Paging, Virtual Memory Concept, Demand Paging, Page Replacement Algorithm, Allocation of Frames, Thrashing

**Chapter 6-File Management**

Introduction, File Structure, Protection, File System Implementation, Directory Structure, Free Space Management, Allocation Methods, Efficiency and Performance, Recovery, NFS and NTFS. SAMBA Concept

**Chapter 7-Disk Management**

Disk Structure, Disk Scheduling Algorithm, Disk Management, Swap Space Concept and Management, RAID Structure, Disk Performance Issues

**Chapter 8-Distributed Operating System**

Introduction, Centralized Versus Distributed Processing, Advantages Of Distributed OS, Types of Distributed OS, Conceptual of Global OS, NOS Architecture

**Chapter 9-Case Study of Window OS and Non-Windows OS**

An Introduction To Modern Mobile Operating Systems, Which Smartphone OS is the best?, Case Study, NON\_WINDOW O.S

=====

**Subject - Object Oriented Programming C++**

CO#	Course Outcome
CO1	Understand the Principle of OOPs and C++
CO2	Understand the Expressions, Functions, Classes and Objects in C++
CO3	Understand the Constructor, Destructor, Operator Overloading and Type Conversion
CO4	Understand the Inheritance, File Handling and advanced C++ topics

## **Chapter-1.Principle of OOP's**

Introduction, What is Object Oriented Development?, Object Oriented Methodology, Overview of procedure Oriented programming, What is Object Oriented Programming?, Object Oriented Languages

## **Chapter-2.Basics of C++**

A Brief History of C and C++, Difference between C and C++, Features of C++, Advantages and Disadvantages of C++, Applications of C++, Writing and Executing a C++ Program, Program Structure and Rules, Sample C++ Program, Comments, Return Type of MAIN(), Namespace std, Header File, Output Statement (COUT), Input Statement (CIN)

## **Chapter-3.Expression**

Introduction, C++ Tokens, Data types , Declaration of Variables, Initialization of Variables, Reference Variables, Operators, Type Cast Operator, Memory Management operators, Expression, Statement, Symbolic Constant, Type Compatibility

## **Chapter-4.Function in C++**

Introduction, Passing Information-Parameters, Default Arguments, Constant Arguments, Function Overloading, Inline Functions, Recursive Functions

## **Chapter-5.Classes and Objects**

Introduction, Class, Member Functions, Making an Outside Function Inline, Nesting Of Member Functions , Private Member Function, Arrays within a Class, Memory Allocation for Objects, Arrays of Objects, Objects as Function Arguments, Returning Objects, Const Member Function, Static Class Members, Pointer to Members, Local classes, Friend Functions, Unions and classes, Object Composition and Delegation

## **Chapter-6.Constructor and Destructor**

Introduction, Constructor, Multiple Constructors in a class, Constructor with Default Arguments, Dynamic Initialization of Objects, Const Object, Destructor

## **Chapter-7.Operator Overloading and Type Conversion**

Introduction, Overloading Unary Operators, Overloading Binary Operators, Limitations of Operator Overloading, this pointer, Overloading <&>Operators, Manipulation of String, Types Conversion

## **Chapter-8.Inheritance**

Introduction, Single Inheritance, multiple Inheritance, Multilevel Inheritance, Hierarchical inheritance, Hybrid Inheritance, Container Classes, Virtual Base Classes, Construction in Derived classes, Virtual Function, Pure Virtual Functions, Abstract Classes

## Chapter-9.The C++ I/O System Basics

Introduction, C++ Streams, C++ Stream Classes, Unformatted I/O Operations, Formatted I/O Operations, Manipulators

## Chapter-10.Working With Files

Introduction, Creating a Stream, Opening a File, Closing a File, Checking For Failure With File Commands, Detecting the End-of-file, file Pointers and their Manipulation, Reading/Writing a character From a File, Write()and read() Functions, Buffers and Synchronization, Other Functions, Random Access File Processing, Updating a File :Random Access, Command Line Arguments

## Chapter-11.Template

Introduction, Generic Functions, A Function with Two Generic Data Types, Explicitly Overloading a generic Function, Overloading Function Templates, using Standard Parameters with Template Function, Generic Functions Restrictions, Generic Class, Using Default Arguments with Template Classes, Template Parameters, Template Specialization, The TypeName and Export Keywords

## chapter-12.Exception Handling

Introduction, the STL Programming Models, Containers, Algorithms, Iterators, Function Objects, Allocators, Adaptors

## Chapter-13.Introduction to Standard Template library

Introduction, The STL Programming Model, Containers, Algorithms, Iterators, Function Objects, Allocators, Adaptors

## Chapter-14.Namespace

Introduction, Defining a Namespace, The Standard Namespace, Nested Namespace, Unnamed Namespace, Namespace Alias

## Chapter-15.New Style Caste and RTRI

Introduction, New-Style Casts, Static cast, Dynamic cast, Const cast, Reinterpret cast, Run-Time Type Information(RTTI), A Simple Application of Run-Time Type ID, Typed can be applied to Templates classes

=====

## Subject – Lab - Object Oriented Programming C++

CO#	Course Outcome
CO1	Understand the Principle of OOPs and C++
CO2	Understand the Expressions, Functions, Classes and Objects in C++
CO3	Understand the Constructor, Destructor, Operator Overloading and Type Conversion
CO4	Understand the Inheritance, File Handling and advanced C++ topics

1. Write a program to find the square of a number.
2. Write a program for Binary to Decimal conversion.
3. Write a program to convert temperature from Fahrenheit to Celsius.
4. Write a program to check whether entered number is Even or Odd using ternary operators.
5. Write a program to reverse an integer.
6. Write a program to add two integers. Make a function add() to add integers and display the sum in main() function.
7. Write a program to print half pyramid using numbers

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

8. Write a program to store and calculate the sum of 5 numbers entered by the user using array.
9. Write a program for addition of members of two different classes using friend Function.
10. Write a program for reading from a text file testout.txt using C++ File Stream programming.