



Course Syllabus: Post Graduate Diploma in Computer Applications

Batch Name: PGDCA

Eligibility: Graduation

Course Start: 1st of Every Month

Course Duration: 350 Hours

Courses / Modules Paper

Module 1: Both the papers are compulsory

Paper 1	Paper 2
Language C	Operating System & Computer Architecture

Module 2: Opt any two papers

Paper 1	Paper 2	Paper 3
Language C++	Data Structures & DBM	Project Work

Module 1:

Paper 1: Language C

1: Introduction of C

- 1.1 History
- 1.2 Why C language
- 1.3 Getting started with C
- 1.4 Writing first program: hello world
- 1.5 Algorithm and flowchart
- 1.6 Understanding the structure and syntax of C
- 1.7 C programming style

2: Fundamentals of C

- 2.1 Introduction
- 2.2 Character set
- 2.3 Keywords
- 2.4 Identifiers
- 2.5 Variables
- 2.6 Data types
- 2.7 Constants
- 2.8 Declaring a keyword
- 2.9 Reading data from keyboard

3: Operators and expressions

- 3.1 Introduction
- 3.2 Arithmetic operator
- 3.3 Increment and Decrement operator
- 3.4 Assignment operator
- 3.5 Relational operator
- 3.6 Logical operator
- 3.7 Conditional operator
- 3.8 Bitwise operator
- 3.9 Special operators

4: Control Statements

- 4.1 Introduction
- 4.2 Branching statements

- If
- Nested if
- If else
- Nested if else
- 4.3 Looping statement
 - For loop
 - While loop
 - Do while loop
- 4.4 Jumping statement
 - Goto
 - Break
 - Continue

5: Function in C

- 5.1 Introduction
- 5.2 Types of functions
- 5.3 Types of user define functions
 - No return type with no arguments
 - Return type with no arguments
 - No return type with arguments
 - Return type with arguments
- 5.4 Need of functions
- 5.5 Scope rules
- 5.6 Recursion
- 5.7 Function using arguments
 - Call by value
 - Call by reference

6: Array in C

- 6.1 Introduction
 - Declaration of Array
 - Initializing an array
 - Accessing an Array
- 6.2 One-dimensional array
- 6.3 Two-dimensional array
 - Nesting of loops
- 6.4 Multi-dimensional array
- 6.5 Dynamic Array

7: Pointers in C

- 7.1 Introduction
- 7.2 Declaration of pointers
- 7.3 Initialization of pointers
- 7.4 Accessing variable through pointer
- 7.5 Pointer to a pointer
 - Declaring a double pointer
- 7.6 Operation on pointer
 - Increment and decrement operator
 - Addition and subtractions of constant value
 - Subtracting two pointers
 - Comparison between two pointers
- 7.7 Pointer to array
- 7.8 Array to pointer
- 7.9 Passing pointer as a function
- 7.10 Pointer to structure
 - Pointer pointing to a function

8: Structures in C

- 8.1 Introduction
- 8.2 Defining a structure
- 8.3 Declaring a structure member
 - Declaring structure variable separately
 - Declaring structure variable along with definition
- 8.4 Accessing structure member
 - With Dot operator
 - With arrow operator
- 8.5 Structure initialization
- 8.6 Array of structure
- 8.7 Using structure as a function argument
 - Passing structure member as an argument
 - Passing structure variable as an argument
 - Passing structures pointer as an argument
- 8.8 Array within structure
- 8.9 Nested structure
- 8.10 typedef keyword
 - Typedef vs #define

9: Union in C

- 9.1 Introduction
- 9.2 Declaration of union
- 9.3 Accessing union member
- 9.4 Bit field

10: Input and output

- 10.1 Introduction
- 10.2 Standard files
- 10.3 Formatted input output
 - scanf()
 - printf()
- 10.4 Unformatted input output
 - getchar()
 - Getch()
 - gets()
 - putchar()
 - puts()
- 10.5 Error Handling
 - errno
 - strerror
 - Perror

11: File I/O

- 11.1 Introduction
- 11.2 Needs of files
- 11.3 Types of files
 - Text files
 - Binary files
- 11.4 Files operations
 - File creation
 - Opening a file
 - Reading data from file
 - Writing data to file
 - Closing a file
- 11.5 Reading writing text files
 - getc()
 - putc()
 - getw()
 - putw()
 - getchar()
 - putchar()
 - fprintf()
 - fscanf()
- 11.6 Reading writing binary files
 - fread()
 - Fwrite()

Paper 2: Operating System & Computer Architecture

Operating System

1: Introduction of OS

- 1.1 Introduction
- 1.2 Basics of OS
- 1.3 Generation of OS
- 1.4 Types of OS
- 1.5 OS services
- 1.6 System calls
- 1.7 OS structure

2: Process Management

- 2.1 Process definition
- 2.2 Process states
- 2.3 Process state transition
- 2.4 Process control block
- 2.5 Context switching
- 2.6 Threads
 - Concepts of multi-threads
 - Benefits of threads
 - Types of threads
- 2.7 Process scheduling Definition
- 2.8 Scheduling Objectives
- 2.9 Types of schedulers
- 2.10 Scheduling Criteria

3: Inter Process Communication

- 3.1 Race condition
- 3.2 Critical section
- 3.3 Mutual Exclusion
- 3.4 Hardware Solution
- 3.5 Strict alternation
- 3.6 Peterson's solution
- 3.7 The producer Consumer Problem

- 3.8 Semaphores
- 3.9 Event Counters
- 3.10 Monitors
- 3.11 Message Passing
- 3.12 Classical IPC Problems

4: Deadlocks

- 4.1 Definition
- 4.2 Deadlock Characteristics
- 4.3 Deadlock prevention
- 4.4 Deadlock avoidance
 - Banker's Algorithm
- 4.5 Deadlock Detection and avoidance

5: Memory Management

- 5.1 Definition
- 5.2 Logical & Physical address map
- 5.3 Memory allocation
- 5.4 Paging
- 5.5 Virtual Memory
 - Basics of virtual memory
 - Hardware & control structure
 - Locality of reference
 - Page Fault
 - Working set
 - Dirty page/ Dirty bit
 - Demand paging
- Page replacement policies

6: Input/ Output Management

- 6.1 Principles of I/O Hardware
 - I/O devices
 - Device controllers
 - Direct memory access
- 6.2 Principles of I/O software
 - Goals of interrupt handlers
 - Device drivers
 - Device independence I/O Software
- 6.3 Secondary-Storage Structure
 - Disk structure
 - Disk scheduling algorithm

7: File Management

- 7.1 File concept
- 7.2 Access method
- 7.3 File types
- 7.4 File Operations
- 7.5 Directory structure
- 7.6 File system structure
- 7.7 Allocation method
- 7.8 Free space management
- 7.9 Directory Implementation

8: Security & Protection

- 8.1 Security environment
- 8.2 Design principles of security
- 8.3 User authentication
- 8.4 Protection Mechanism
- 8.5 Protection Domain
- 8.6 Access Control List

Computer Architecture

1: Boolean Algebra

- 1.1 Definition
- 1.2 Properties of Boolean algebra
- 1.3 Minterm & Maxterm
- 1.4 Logic operations
- 1.5 Digital Logic gates
- 1.6 IC digital Logic families

2: Basic gates computations

- 2.1 Different types of Flip-Flop
- 2.2 Product of sum simplifications NAND or NOR implementation
- 2.3 Don't care condition
- 2.4 Tabulation method
- 2.5 Adder
- 2.6 Subtractor
- 2.7 Code Conversion
- 2.8 Universal gate

3: Sequential Logics

- 3.1 Flip-Flops
- 3.2 Triggering of Flip-Flop
- 3.3 Analysis of clocked sequential circuits
- 3.4 Flip-Flop Excitation
- 3.5 Design with static equation

4: Basic Computer Organization & Design

- 4.1 Instruction Codes, Computer registers
- 4.2 Computer instructions
- 4.3 Timing & Control, Instruction cycle
- 4.4 Memory - Reference Instruction
- 4.5 Input-Output & interrupt
- 4.6 Design of basic computer

5: The basic computer

- 5.1 Introduction
- 5.2 Machine Language
- 5.3 Assembly Language
- 5.4 Program Loops
- 5.5 Subroutines
- 5.6 IO Programming

Module 2:

Paper 1: Language C++

1: Introduction of C++

- 1.1 History
- 1.2 Why Java language
- 1.3 Getting started with C++
- 1.4 Writing first program: hello world
- 1.5 Algorithm and flowchart
- 1.6 Understanding the structure and syntax of C++
- 1.7 C++ programming style

2: Principle of OOP

- 2.1 Introduction to OOP
- 2.2 OOP vs Procedural oriented programming
- 2.3 objects
- 2.4 Classes
- 2.5 Encapsulation
- 2.6 Abstraction
- 2.7 Polymorphism
- 2.8 Inheritance
- 2.9 Applications of OOP

3: Operators in C++

- 3.1 Introduction to operators
- 3.2 Special operators
- 3.3 scope resolution operator
- 3.4 Member dereferencing operator
- 3.5 Memory management operator
- 3.6 Manipulators and Typecast operator

4: variables and Functions in C++

- 4.1 Introduction
- 4.2 Functions
 - Devlaration
 - Definition
- 4.3 Variables
 - Declaration
 - Scope of variables
- 4.4 Arrays and Strings

5: Classes and Objects

- 5.1 Introduction
- 5.2 Declaring a Class
 - Data members
 - Member function
 - Private and public members
 - Accessing member functions
- 5.3 Functions
 - Class function definition
 - Member function definition
- 5.4 Objects
 - Creating an object
 - Accessing members using objects
- 5.5 Object as a function argument
 - Pass by value
 - Pass by reference

6: Operator Overloading

- 6.1 Introduction
- 6.2 Operator Overloading Methods
- 5.3 Type Conversion

7: Constructors in C++

- 7.1 Introduction
- 7.2 Declaration of Constructor
- 7.3 Definition of Constructor
- 7.4 Types of Constructor
 - Default Constructor
 - Parameterized Constructor
 - Copy Constructor
- 7.5 Destructor
 - Definition
 - Use of Destructor

8: Inheritance

- 8.1 Introduction
- 8.2 Base Class and Derived Class
- 8.3 Visibility Mode
 - Private
 - Public
 - Protected
- 8.4 Types of Inheritance
 - Single inheritance
 - Multilevel inheritance
 - Multiple Inheritance
 - Hierarchal Inheritance
 - Hybrid inheritance
- 8.5 Nesting of Classes

9: Polymorphism

- 9.1 Introduction
- 9.2 Application and Demonstration
- 9.3 Early Binding
- 9.4 Polymorphism with pointers
- 9.5 Early Binding
- 9.6 Late Binding
- 1.7 Virtual Functions
- 9.8 Pure Virtual Functions

10: Exception Handling

- 10.1 Introduction
- 10.2 Exception handling mechanism
 - Throw an Exception
 - Catch an Exception
 - Rethrow an Exception

11: Input and Output in C++

- 11.1 Introduction
- 11.2 Standard Streams
- 11.3 Manipulators
- 11.4 Unformatted Input
- 11.5 Unformatted Output

12: File Processing

- 12.1 Introduction
- 12.2 Opening and Closing of file
- 12.3 Binary File Operations
- 12.4 Structures and file operations
- 12.4 Classes and File operations
- 12.5 Random File Processing

Paper 2: Data structure & DBMS**Data structure****1: Data structure & Algorithm**

- 1.1 Introduction
- 1.2 Asymptotic Analysis
- 1.3 DS Pointers
- 1.4 DS Structures
- 1.5 Algorithm Definition
- 1.6 Characteristics of algorithm
- 1.7 Elements of algorithm

2: Functions

- 2.1 what is function
- 2.2 Types of function
- 2.3 how functions works
- 2.4 functions recursion & how work

3: Arrays

- 3.1 Concepts of array
- 3.2 Types of array
- 3.3 Basic Programs
- 3.4 Array with functions
- 3.5 Types of array

4: Pointers

- 4.1 Pointer Basics
- 4.2 Pointer with functions

- 4.3 Call by reference
- 4.4 Array of pointers & pointer to Array & Programs

5: Structure

- 5.1 Understanding about Structure
- 5.2 Pointer structure variable
- 5.3 Structure as a function Argument
- 5.4 Using call by value
- 5.5 Passing reference of structure

6: Stacks

- 6.1 Operations on stacks
- 6.2 Arrays & linked representation
- 6.3 Programs on stacks
- 6.4 Push & Pop operations
- 6.5 Traversing

7: Applications of stacks

- 7.1 Arithmetic, Expression Evaluation
- 7.2 Notations
- 7.3 Conversions
- 7.4 Evaluations

8: Queue

- 8.1 Operations on Queue
- 8.2 Array & linked representation
- 8.3 Programs on stacks
- 8.4 Insert & Delete operations
- 8.5 Circular queue
- 8.6 Applications of queue

9: Linked List

- 9.1 Concept of linked list
- 9.2 Difference b/w Linked list & Array
- 9.3 Single linked list
- 9.4 Representation
- 9.5 Operations
- 9.6 Traversing
- 9.7 Insertion
- 9.8 Deletion
- 9.9 Double linked list
- 9.10 Representation
- 9.11 Operations, Traversing
- 9.12 Circular linked list

10: Trees

- 10.1 Introduction
- 10.2 Binary Tree
- 10.3 Complete binary tree
- 10.4 Binary Search tree
- 10.5 Tree traversal
- 10.6 Expression tree
- 10.7 Binary search tree
- 10.8 AVL tree

11: Graph

- 11.1 Graph terminology
- 11.2 Representation of graphs
- 11.3 Path matrix
- 11.4 Graph Traversal
- 11.5 BFS (*breadth first search)
- 11.6 DFS (depth first search)
- 11.7 Minimum spanning tree
- 11.8 Kruskal's algorithm & prim's Algorithm

12: Hashing & Searching

- 12.1 Linear & Binary search trees
- 12.2 Hash function
- 12.3 Hashing techniques & chaining

13: Sorting

- 13.1 Bubble sort
- 13.2 Selection sort
- 13.3 Insertion sort
- 13.4 Quick sort
- 13.5 Merge sort
- 13.6 Heap sort
- 13.7 Radix sort

DBMS**1: Introduction to DBMS**

- 1.1 What is database system
- 1.2 Purpose of DBMS
- 1.3 View of data
- 1.4 Relational databases
- 1.5 Database architecture

2: Data Models

- 2.1 Introduction
- 2.2 Importance of data model
- 2.3 Basic building blocks
- 2.4 ?Business rules
- 2.5 The evolution of data model
- 2.6 Degree of data abstraction

3: Database Design

- 3.1 Database design & ER model Overview
- 3.2 Constraints
- 3.3 ER Diagram
- 3.4 Weak entity set
- 3.5 Codd's rule
- 3.6 Relational schemes
- 3.7 Relational database model
 - Logical view of data
 - Keys
 - Integrity rules
- 3.8 Relational Database design
 - Features of good relational database design
 - Atomic
 - Domain and Normalization (1NF, 2NF, 3NF, BCNF).

4: Relational algebra & Calculus

- 4.1 Introduction
- 4.2 Selection & Projection
- 4.3 Set operations
- 4.4 Renaming
- 4.5 Joins
- 4.6 Syntax, Semantics, Operations
- 4.7 Grouping & Ungrouping
- 4.8 Relational Comparison
- 4.9 Calculus
 - Tuples relational calculus
 - Domain relational calculus
 - Calculus vs algebra
 - Computational capabilities

5: Constraints, Views & SQL

- 5.1 What is Constraints
- 5.2 Types of constraints
- 5.3 Integrity constraints
- 5.4 View
 - Introduction
 - Data independence
 - Security
 - Updates on Views
 - Comparison b/w tables & Views
- 5.5 SQL
 - Data definition
 - Aggregate Function
 - Null values
 - Nested sub queries
 - Joined relations, Triggers

Paper 3: Project Work

Student have to submit project on the basis of concerned subjects.