14. COMPUTER SCIENCE (Code 083)

Learning Objectives:

1. To develop logic for Problem Solving
2. To understand the concept of Object Oriented Methodology
3. To implement Object Oriented Programming using C++
4. To understand the concept of working with Relational Database
5. To understand the basic concept of Logic of Computing
6. To understand the basic concepts of Communication and Networking technologies
7. To understand Open Source Software

Competencies:

The student will develop the following proficiency:

1. Identifying Computer Components / Subsystems / Peripherals
2. Problem Solving using Object Oriented Programming
3. Database Handling

Class XI (Theory)

Duration: 3 hours **Total Marks: 70**

Unit No.	Unit Name	Periods			Marks		
		Th	Pr	Tot	Th	Pr	Total
1.	COMPUTER FUNDAMENTALS	10	5	15	10	2	12
2.	INTRODUCTION TO C++	25	20	45	14	8	22
3.	PROGRAMMING METHODOLOGY	10	10	20	10	2	12
4.	PROGRAMMING IN C++	65	35	100	36	18	54
		110	**70**	**180**	**70**	**30**	**100**

UNIT 1: COMPUTER FUNDAMENTALS

Evolution of computers; Basics of computer system and its operation: Functional Components and their inter-connections; concept of Booting.

Software Concepts:

Types of Software - System Software, Utility Software and Application Software;

System Software: Operating System, Compiler, Interpreter and Assembler;

Operating System: Need for operating system, Functions of Operating System (Processor Management, Memory Management, File Management and Device Management), Types of operating system -

Interactive (GUI based), Real Time and Distributed; Commonly used operating systems: UNIX, LINUX, Windows, Solaris, BOSS (Bharat Operating System Solutions); Mobile OS - Android, Symbian.

Illustration and practice of the following tasks using any one of the above Operating Systems:

- Opening/Closing Windows
- Creating/Moving/Deleting Files/Folders
- Renaming Files/Folders
- Switching between Tasks

Utility Software: Anti Virus, File Management tools, Compression tools and Disk Management tools (Disk Cleanup, Disk Defragmenter, Backup)

Application software: Office Tools - Word Processor, Presentation Tool, Spreadsheet Package, Database Management System; Domain specific tools - School Management System, Inventory Management System, Payroll System, Financial Accounting, Hotel Management, Reservation System and Weather Forecasting System

Number System: Binary, Octal, Decimal, Hexadecimal and conversion amongst these number systems.

Internal Storage encoding of Characters: ASCII, ISCII (Indian scripts Standard Code for Information Interchange), and UNICODE (for multilingual computing)

Microprocessor: Basic concepts, Clock speed (MHz, GHz), 16 bit, 32 bit, 64 bit processors, 128 bit processors; Types - CISC Processores (Complex Instruction set computing), RISC Processors (Reduced Instruction set computing), and EPIC (Explicitly parallel Instruction computing).

Memory Concepts:

Units: Byte, Kilo Byte, Mega Byte, Giga Byte, Tera Byte, Peta Byte, Exa Byte, Zetta Byte, Yotta Byte

Primary Memory: Cache, RAM, ROM

Secondary Memory: Fixed and Removable Storage - Hard Disk Drive, CD/DVD Drive, Pen Drive, Blue Ray Disk

Input Output Ports/Connections: Serial, Parallel and Universal Serial Bus, PS-2 port, Infrared port, Bluetooth, Firewire.

Note : Exploring inside computer system in the computer lab class.

UNIT 2: INTRODUCTION TO C++

Getting Started :

C++ character set, C++ Tokens (Identifiers, Keywords, Constants, Operators), Structure of a C++ Program (include files, main function), Header files - iostream.h, iomanip.h, **cout, cin;** Use of I/O operators (<< and >>), Use of endl and setw (), Cascading of I/O operators, Error Messages; Use of editor, basic commands of editor, compilation, linking and execution.

Data Types, Variables and Constants:

Concept of Data types; Built-in Data types: **char, int, float** and **double;** Constants: Integer Constants, Character constants - \n, \t, \b), Floating Point Constants, String Constants; Access modifier: **const;** Variables of built-in data types, Declaration/Initialisation of variables, Assignment statement; Type modifier: **signed, unsigned, long**

Operator and Expressions :

Operators: Arithmetic operators (-,+,*,/,%), Unary operator (-), Increment (++) and Decrement (--) Operators, Relation operator (>,>=,<,<=,= =,!=), Logical operators (!, &&,||), Conditional operator: <condition>? <if true>:<if false>; Precedence of Operators; Automatic type conversion in expressions, Type casting; C++ shorthands (+=, -=, *=, /=, %=)

UNIT 3: PROGRAMMING METHODOLOGY

General Concepts; Modular approach; Clarity and Simplicity of Expressions, Use of proper Names for identifiers, Comments, Indentation; Documentation and Program Maintenance; Running and Debugging programs, Syntax Errors, Run-Time Errors, Logical Errors

Problem Solving Methodologies: Understanding of the problem, Identifying minimum number of inputs required for output, Writing code to optimizing execution time and memory storage, step by step solution for the problem, breaking down solution into simple steps, Identification of arithmetic and logical operations required for solution, Control Structure: Conditional control and looping (finite and infinite)

UNIT 4: PROGRAMMING IN C++

Flow of control:

Conditional statements: **if-else**, Nested **if**, **switch..case..default**, use of conditional operator, Nested **switch..case, break** statement (to be used in **switch..case only**); Loops: **while, do - while , for** and Nested loops

Inbuilt Functions

Header file Categorization	Header File	Function
Standard input/output functions	stdio.h	gets ( ), puts ( )
Character Functions	ctype.h	isalnum ( ), isalpha ( ), isdigit ( ), islower ( ), isupper ( ), tolower ( ), toupper ( )
String Functions	string.h	strcpy ( ), strcat ( ), strlen ( ), strcmp ( ), strcmpi ( ), strrev ( ), strlen ( ), strupr ( ), strlwr ( )
Mathematical Functions	math.h	fabs ( ), pow ( ), sgrt ( ), sin ( ), cos ( ), abs ( )
Other Functions	stdlib.h	randomize ( ), random ( ), itoa ( ), atoi ( )
User Defined Functions:		

Introduction to user-defined function and its requirements.

Defining a function; function prototype, Invoking/calling a function, passing arguments to function, specifying argument data types, default argument, constant argument, call by value, call by reference, returning values from a function, calling functions with arrays, scope rules of functions and variables local and global variables.

Relating the Parameters and return type concepts in built-in functions.

Structured Data Type:

Arrays: Introductory to Array and its advantages.

One Dimensional Array : Declaration/initialisation of One-dimensional array, Inputting array elements, Accessing array elements, Manipulation of Array elements (sum of elements, product of elements, average of elements, linear search, finding maximum/minimum value)

Declaration/Initialization of a String, string manipulations (counting vowels/ consonants/digits/special characters, case conversion, reversing a string, reversing each word of a string)

Two-dimensional Array

Declaration/initialisation of a two-dimensional array, inputting array elements Accessing array elements, Manipulation of Array elements (sum of row element, column elements, diagonal elements, finding maximum/minimum values)

User-defined Data Types:

Introduction to user defined data types.

Structure

Defining a Structure (Keyword Structure), Declaring structure variables, Accessing structure elements, Passing structure to Functions as value and reference argument/parameter, Function returning structure, Array of structures, passing an array of structure as an argument/ a parameter to a function

Defining a symbol name using **typedef** keyword and defining a macro using **#define** directive.

Class XI (Practical)

Duration: 3 hours **Total Marks: 30**

1. Programming in C++ **10**

One programming problem in C++ to be developed and tested in Computer during the examination. Marks are allotted on the basis of following:

Logic	:	5 Marks
Documentation/Indentation	:	2 Marks
Output presentation	:	3 Marks

2 Project Work **06**

Problems related to String, Number and Array manipulation

General Guidelines: Initial Requirement, developing an interface for user (it is advised to use text based interface screen), developing logic for playing the game and developing logic for scoring points

1. Memory Game: A number guessing game with application of 2 dimensional arrays containing randomly generated numbers in pairs hidden inside boxes.
2. Cross 'N Knots Game: A regular tic-tac-toe game
3. Hollywood/Hangman: A word Guessing game
4. Cows 'N Bulls: A word/number Guessing game

or

Similar projects may be undertaken in other domains

(As mentioned in general guidelines for project, given at the end of the curriculum in a group of 2-4 students)

3. Presentation based on research **02**

It will be a group presentation based on a detailed study of at least two technology inventions in the field of information technology, which may include Inventor's name with country, out of box contributions year of invention, characteristics, social impact and uses. A partial list of inventors is in the Annexure.

(The project can be done in a group of 2-3 students)

4 Practical File **06**

(a) Record of the configuration of computer system used by the student in the computer lab (by exploring inside computer system in the first 2 lab classes).

(b) Must have minimum 15 programs from the topics covered in class XI course.

- 5 Programs on Control structures
- 4 Programs on Array manipulations
- 4 Programs on String Manipulations
- 2 Programs on structure manipulations

5 Viva Voce **06**

Viva will be asked from the syllabus covered in class XI and the project developed by the student(s).

Class XII (Theory)