

Computer Science (Code 083)

“...It is unworthy of excellent men to loose hours like slaves in the labour of calculation which could safely be regulated to anyone else if machines were used..” said Leibnitz in the beginning of seventeenth Century.

Farsighted vision, in-depth study accompanied by target-oriented effort of such torchbearers have ushered in an age of computers.

Be it Science or Engineering, medical world or launching Space Shuttles, Study of Universe or global communications, Research and Development or Edutainment – the core ingredient is computer.

Learning Objectives:

1. To understand the problem statement
2. To develop logic for problem solving
3. To understand the concept of Object Oriented Methodology
4. To implement Object Oriented Programming using C++
5. To understand the concept of working with Relational Database
6. To understand the basic concept of algebra of logic
7. To understand and explore the world of communication and networks

Competencies:

The student will be proficient in the following:

1. Identification of a system
2. Categorisation of parts of an objective system
3. Problem Solving
4. Designing an efficient logic using object oriented approach for solution development
5. Database handling
6. Logic Circuit designing

Class XI (Theory)

Duration: 3 hours

Total Marks: 70

Unit No.	Unit Name	Marks
1.	COMPUTER FUNDAMENTALS	06
2.	PROGRAMMING METHODOLOGY	10
3.	INTRODUCTION TO PROGRAMMING IN C++	44
4.	COMPUTER SYSTEM ORGANISATION	10
		70

UNIT 1: COMPUTER FUNDAMENTALS

Evolution of computers; Basics of computer and its operation: Functional Components and their inter-connections, concept of Booting, Use of Operating System for directory listing, hierarchical directory structure, renaming, deleting files/folders, formatting floppy, copying files, concepts of path and pathname, switching between tasks, installation/removal of applications;

Computer Science (Code 083)

Software Concepts:

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

System Software: Operating System, Compilers, Interpreters 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), Time Sharing, Real Time and Distributed;

Note: The above-mentioned Operating System specific tasks can be illustrated and implemented using any operating system.

UNIT 2: 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 Methodology and Techniques: Understanding of the problem, Identifying minimum number of inputs required for output, Step by step solution for the problem, breaking down solution into simple steps, Identification of arithmetic and logical operations required for solution, Using Control Structure: Conditional control and looping (finite and infinite);

UNIT 3: INTRODUCTION TO PROGRAMMING IN C++

“Object Oriented Technology is regarded as the ultimate paradigm for the modeling of information, be that information data or logic. The C++ has by now shown to fulfill this goal.”

Programming by Example In C++ Language :

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; standard input/output operations from C language: gets(), puts() of stdio.h header file;

Data Types, Variables and Constants:

Concept of Data types; Built-in Data types: **char**, **int**, **float** and **double**; Constants: Integer Constants, Character Constants (Backslash character constants - /n, /t), 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;;

Operators and Expressions:

Computer Science (Code 083)

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

Flow of control:

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

Structured Data Type: 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/initialisation of a String, string manipulations (counting vowels/consonants/digits/special characters, case conversion, reversing a string, reversing each word of a string); String and Character related Library functions: **isalnum()**, **isalpha()**, **isdigit()**, **islower()**, **isupper()**, **tolower()**, **toupper()**, **strcpy()**, **strcat()**, **strlen()**, **strcmp()**, **strcmpi()**;

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 Functions:

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;

Mathematical and Other Functions:

Header Files-math.h, stdlib.h;

Functions: **fabs()**, **log()**, **log10()**, **pow()**, **sqrt()**, **sin()**, **cos()**, **abs()**, **randomize()**, **random()** ;

Event programming: Games as examples

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

Computer Science (Code 083)

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

UNIT 4: COMPUTER SYSTEM ORGANISATION

Number System: Binary, Octal, Decimal, Hexadecimal and conversion between two different number systems. Integer, Floating Point, 2's complement of number from base-2;

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

Microprocessor: Basic concepts, Clock speed (MHz, GHz), 16 bit, 32 bit, 64 bit processors; Types – CISC, RISC; Concept of System Buses, Address bus, Data bus,

Concepts of Accumulator, Instruction Register, and Program Counter;

Commonly used CPUs and CPU related terminologies: Intel Pentium Series, Intel Celeron, Cyrix, AMD Series, Xeon, Intel Mobile, Mac Series; CPU Cache; Concept of heat sink and CPU fan, Motherboard; Single, Dual and Multiple processors;

Types of Memory: Cache (L1,L2), Buffer, RAM (DRAM, SDRAM, RDRAM, DDRAM), ROM (PROM, EPROM), Access Time;

Input Output Ports/Connections: Power connector, Monitor Socket, Serial (COM) and Parallel (LPT) port, Universal Serial Bus port, PS-2 port, SCSI port, PCI/MCI socket, Keyboard socket, Infrared port (IR), audio/speaker socket, Mic socket; data Bus; external storage devices connected using I/O ports;

Power Supply: Switched Mode Power Supply (SMPS): Elementary Concept of Power Supply: Voltage, Current, Power (Volt, Ampere, Watt), SMPS supplies – Mother Board, Hard Disk Drive, Floppy Disk Drive, CD/DVD Drive;

Power Conditioning Devices: Voltage Stabilizer, Constant Voltage Transformer (CVT), Uninterrupted Power Supply (UPS)-Online and offline.

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

Notes: The types of problems to be given will be of application type from the following topics

2 Project Work **10**

(As mentioned in general guidelines for project, given at the end of the curriculum)

3 Practical File **05**

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

4 Viva Voce **05**

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

Class XII (Theory)

Duration: 3 Hours

Total Marks: 70

Unit No.	Unit Name	Marks
1.	PROGRAMMING IN C++	30
2.	DATA STRUCTURE	16
3.	DATABASE AND SQL	8
4.	BOOLEAN ALGEBRA	8
5.	COMMUNICATION AND NETWORK CONCEPTS	8
		70

UNIT 1: PROGRAMMING IN C++

REVIEW: C++ covered In Class -XI,
Defining a symbol name using typedef keyword and defining a macro using #define directive;

Need for User defined data type;

Structures:

Defining a 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;

Object Oriented Programming:

Concept of Object Oriented Programming – Data hiding, Data encapsulation, Class and Object, Abstract class and Concrete class, Polymorphism (Implementation of polymorphism using Function overloading as an example in C++); Inheritance, Advantages of Object Oriented Programming over earlier programming methodologies,

Implementation of Object Oriented Programming concepts in C++:

Definition of a class, Members of a class - Data Members and Member Functions (methods), Using Private and Public visibility modes, default visibility mode (private); Member function definition: inside class definition and outside class definition using scope resolution operator (::); Declaration of objects as instances of a class; accessing members from object(s), Array of type class, Objects as function arguments - pass by value and pass by reference;

Constructor and Destructor:

Constructor: Special Characteristics, Declaration and Definition of a constructor, Default Constructor, Overloaded Constructors, Copy Constructor, Constructor with default arguments;

Destructor: Special Characteristics, Declaration and definition of destructor;

Computer Science (Code 083)

Inheritance (Extending Classes): Concept of Inheritance, Base Class, Derived Class, Defining derived classes, protected visibility mode; Single level inheritance, Multilevel inheritance and Multiple inheritance, Privately derived, Publicly derived and Protectedly derived class, accessibility of members from objects and within derived class(es);

Data File Handling:

Need for a data file, Types of data files – Text file and Binary file;

Basic file operations on text file: Creating/Writing text into file, Reading and Manipulation of text from an already existing text File (accessing sequentially);

Binary File: Creation of file, Writing data into file, Searching for required data from file, Appending data to a file, Insertion of data in sorted file, Deletion of data from file, Modification of data in a file;

Implementation of above mentioned data file handling in C++;

Components of C++ to be used with file handling:

Header file: fstream.h; ifstream, ofstream, fstream classes;

Opening a text file in **in**, **out**, and **app** modes;

Using cascading operators for writing text to the file and reading text from the file; **open()**, **get()**, **put()**, **getline()** and **close()** functions; Detecting end-of-file (with or without using **eof()** function);

Opening a binary file using **in**, **out**, and **app** modes;

open(), **read()**, **write()** and **close()** functions; Detecting end-of-file (with or without using **eof()** function); **tellg()**, **tellp()**, **seekg()**, **seekp()** functions

Pointers:

Declaration and Initialization of Pointers; Dynamic memory allocation/deallocation operators: **new**, **delete**; Pointers and Arrays: Array of Pointers, Pointer to an array (1 dimensional array), Function returning a pointer, Reference variables and use of alias; Function call by reference. Pointer to structures: Deference operator: *, ->; self referencial structures;

UNIT 2: DATA STRUCTURES

Arrays:

One and two Dimensional arrays: Sequential allocation and address calculation;

One dimensional array: Traversal, Searching (Linear, Binary Search), Insertion of an element in an array, deletion of an element from an array, Sorting (Insertion, Selection, Bubble sort), concatenation of two linear arrays, merging of two sorted arrays;

Computer Science (Code 083)

Two-dimensional arrays: Traversal, Finding sum/difference of two NxM arrays containing numeric values, Interchanging Row and Column elements in a two dimensional array;

Stack (Array and Linked implementation of Stack):

Operations on Stack (PUSH and POP) and its Implementation in C++, Converting expressions from INFIX to POSTFIX notation and evaluation of Postfix expression;

Queue: (Circular Array and Linked Implementation):

Operations on Queue (Insert and Delete) and its Implementation in C++.

UNIT 3: DATABASES AND SQL

Database Concepts:

Relational data model: Concept of domain, tuple, relation, key, primary key, alternate key, candidate key;

Relational algebra: Selection, Projection, Union and Cartesian product;

Structured Query Language:

General Concepts: Advantages of using SQL, Data Definition Language and Data Manipulation Language;

Data types: NUMBER, CHARACTER, DATE;

SQL commands:

CREATE TABLE, DROP TABLE, ALTER TABLE, UPDATE...SET..., INSERT, DELETE;

SELECT, DISTINCT, FROM, WHERE, IN, BETWEEN, GROUP BY, HAVING, ORDER BY;

SQL functions: SUM, AVG, COUNT, MAX and MIN;

Note: Implementation of the above mentioned commands could be done on any SQL supported software.

UNIT 4: BOOLEAN ALGEBRA

Evolution of Boolean algebra, Binary-valued Quantities, Boolean Variable, Boolean Constant and Boolean Operators: AND, OR, NOT; Truth Tables; Closure Property, Commutative Law, Associative Law, Identity law, Inverse law, Principle of Duality, Idempotent Law, Distributive Law, Absorption Law, Involution law, DeMorgan's Law and their applications;

Computer Science (Code 083)

Obtaining Sum of Product (SOP) and Product of Sum (POS) form from the Truth Table, Reducing Boolean Expression (SOP and POS) to its minimal form, Use of Karnaugh Map for minimisation of Boolean expressions (up to 4 variables);

Basic Logic Gates (NOT, AND, OR, NAND, NOR) and their use in circuits.

UNIT 5: COMMUNICATION AND NETWORK CONCEPTS

Evolution of Networking: ARPANET, Internet, Interspace;

Different ways of sending data across the network with reference to switching techniques;

Data Communication terminologies: Concept of Channel, Baud, Bandwidth (Hz, KHz, MHz) and Data transfer rate (bps, kbps, Mbps, Gbps, Tbps);

Transmission media: Twisted pair cable, coaxial cable, optical fiber, infrared, radio link, microwave link and satellite link.

Network devices: Modem, RJ45 connector, Ethernet Card, Hub, Switch, Gateway;

Different Topologies- Bus, Star, Tree; Concepts of LAN, WAN, MAN;

Protocol: TCP/IP, File Transfer Protocol (FTP), PPP, Level-Remote Login (Telnet), Internet, Wireless/Mobile Communication, GSM, CDMA, WLL, 3G, SMS, Voice mail, Application Electronic Mail, Chat, Video Conferencing;

Network Security Concepts: Cyber Law, Firewall, Cookies, Hackers and Crackers;

WebPages; Hyper Text Markup Language (HTML), eXtensible Markup Language (XML); Hyper Text Transfer Protocol (HTTP); Domain Names; URL; Protocol Address; Website, Web browser, Web Servers; Web Hosting.

Class XII (Practicals)

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

Notes: The types of problems to be given will be of application type from the following topics

- Arrays (One dimensional and two dimensional)
- Array of structure
- Stack using arrays and linked implementation
- Queue using arrays (circular) and linked implementation
- Binary File operations (Creation, Displaying, Searching and modification)
- Text File operations (Creation, Displaying and modification)

2. SQL Commands **05**

Five Query questions based on a particular Table/Relation to be tested practically on Computer during the examination. The command along with the result must be written in the answer sheet.

3. Project Work **05**

The project has to be developed in C++ language with Object Oriented Technology and also should have use of Data files.

- Presentation on the computer
- Project report (Listing, Sample, Outputs, Documentation)
- Viva

4. Practical File **05**

Must have minimum 20 programs from the following topics

- Arrays (One dimensional and two dimensional, sorting, searching, merging, deletion & insertion of elements)
- Arrays of structures, Arrays of Objects
- Stacks using arrays and linked implementation
- Queues using arrays (linear and circular) and linked implementation

Computer Science (Code 083)

- File (Binary and Text) operations (Creation, Updation, Query)
- Any computational based problems

15 SQL commands along with the output based on any table/relation: 3 Marks

5. Viva Voce 05

Viva will be asked from syllabus covered in class XII and the project developed by student.

GUIDELINES FOR PROJECTS (Class XI and XII)

1. Preamble

1.1 The academic course in Computer Science includes one Project in each year. The Purpose behind this is to consolidate the concepts and practices imparted during the course and to serve as a record of competence.

1.2 A group of two students/three student as team may be allowed to work on one project.

2. Project content

2.1 Project for class XI can be selected from one of the topics given in event programming.

2.2 Project for class XII should ensure the coverage of following areas of curriculum:

- a. Problem Solving
- b. Data Structure
- c. Object Oriented Programming in C++
- d. Data File Handling

Theme of the project can be

- Any subsystem of a System Software or Tool
- Any Scientific or a fairly complex algorithmic situation.
- Business oriented problems like Banking, Library information system, Hotel or Hospital management system, Transport query system
- Quizzes/Games;
- Tutor/Computer Aided Learning Systems

2.3 The aim of the project is to highlight the abilities of algorithmic formulation, modular programming, optimized code preparation, systematic documentation and other associated aspects of Software Development.

Computer Science (Code 083)

- 2.4 The assessment would be through the project demonstration and the Project Report, which should portray Programming Style, Structured Design, Minimum Coupling, High Cohesion, Good documentation of the code to ensure readability and ease of maintenance.

Computer Science (Code 083)

Reference Books

Computer Organisation and boolean Algebra

1. Rajaraman, FUNDAMENTALS OF COMPUTERS, Prentice Hall of India.
2. C.W.Gear, COMPUTER ORGANISATION AND PROGRAMMING, McGraw Hill Publishing.
3. A.P. Malvino, DIGITAL COMPUTER FUNDAMENTALS, Tata McGraw Hill Publishing Co. Ltd.
4. J. Shelly & Roger Hunt, COMPUTER STUDIES, Wheeler's Publication.
5. C.S. French, COMPUTER STUDIES, Arnold Publishers.
6. Thomas C. Bartee, DIGITAL COMPUTER FUNDAMENTALS, McGraw Hill International.

Problem Solving and Programming in C++

Note: Prior knowledge of C is not required in the learning of C++, eventhough reference about C are made in some of the books.

1. Robert Lofore, OBJECT ORIENTED PROGRAMMING IN TURBO C++, Galgotia Publications Pvt. Ltd.
2. David Parsons, OBJECT ORIENTED PROGRAMMING WITH C++, BPB Publications.
3. Bjarne Stroustrup, THE C++ PROGRAMMING LANGUGE, Adison Wesley.
4. Al Stevens, TEACH YOUR SELF C++ TECHNIQUES & APPLICATIONS, BPB Publications.
5. Scott Robbert Ladd, TURBO C++ TECHNIQUES & APPLICATIONS, BPB Publications.

Operating Environment

1. Ritchi, Operating Systems, BPB Publications.
2. James L. Peterson & Abraham S., OPERATING SYSTEM, Addison-Wesley Publishing Company.

Data Structures

1. M.A., Weiss, Data Structures and Algorithm Analysis in C++. the Benjamin/Cummings Pub. Co., Inc.
2. Scott Rober Ladd, C++ COMPONENTS AND ALGORITHMS, BPB Publications.

Database Management System and SQL

1. Martin Gruber, UNDERSTANDING SQL, BPB Publications.
2. Sheldon M. Dunn x Base Cross Reference Handbook, First Authorised Asian Edition 93, Tech. Publications Pvt. Ltd.
3. C.J. Data, DATABASE PRIMER, Adison Wesley.

Computer Network

1. A.S. Tanenbaum, Computer Network, Prentice Hall of India P. Ltd.
2. Williams Stalling, Data Communication and Networks, Prentice Hall of

Computer Science (Code 083)

- India P. Ltd.
3. Hancock, Network Concept and Architectures, BPB Publications.

Reference Magazines

PC WORLD, COMPUTERS TODAY, PC QUEST, DATA QUEST, COMPUTER WORLD.

Reference Manuals

OPERATING SYSTEM MANUAL, C++ COMPILER MANUAL