Computer Science Class 11th Syllabus

Course Structure

Unit	Title	Marks
1.	Computer Fundamentals	10
2.	Programming Methodology	12
3.	Introduction to Python/C++	18/14
4.	Programming with Python/C++	30/34
	Total	**70**

Unit 1: Computer Fundamentals

Classification of computers: basics of computer and its operation; functional components and their interconnections, concept of booting.

Software concepts: Types of Software - system software, utility software and application software

System Software: Operating system, complier, 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), time sharing, real time and distributed, commonly used operating system: UNIX, LINUX, Windows, Solaris, BOSS (Bharat Operating System Solutions); Mobile OS –Android, Symbian.

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

Open Source Concepts: Open source software, freeware, shareware, proprietary software.

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 between two different 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, 128 bit processors; Types - CISC Processors (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.

Unit 2: Programming Methodology

General Concepts: 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, solution for the problem, breaking down solution into simple steps (modular approach), identification of arithmetic and logical operations required for solution; control structure- conditional control and looping (finite and infinite).

Problem Solving: Introduction to algorithms/flowcharts.

Unit 3: Introduction to Python

Getting Started: Introduction to Python - an integrated high level language, interactive mode and script mode. Data types –Number (Integer - boolean, decimal, octal, hexadecimal; Floating point; Complex), none, Sequence (String,Tuples, List) Sets, Mapping.

Mutable and Immutable Variables

Variables, Expressions and Statements: Values, Variables and keywords; Operators and Operands in Python: (Arithmetic, relational and logical operators), operator precedence, Expressions and Statements (Assignment statement); Taking input (using raw_input() and input()) and displaying output (print statement); Putting Comments.

Functions: Importing Modules (entire module or selected objects), invoking built in functions, functions from math module (for example, ceil, floor, fabs, exp, log, log10, pow, sqrt, cos, sin, tan, degrees, radians), using random() and randint() functions of random module to generate random numbers, composition.

Defining functions, invoking functions, passing parameters (default parameter values, keyword arguments), scope of variables, void functions and functions returning values, flow of execution

Conditional constructs and looping: if else statement while, for (range function), break, continue, else, pass, nested if, nested loops, use of compound expression in conditional and looping construct.

Unit 4: Programming with Python

Strings: Creating, initialising and accessing the elements; string operators: +, *, in, not in, range slice [n:m]; comparing strings using relational operators; String functions & methods: len, capitalize, find, isalnum, isalpha, isdigit, lower, islower, isupper, upper, lstrip, rstrip, isspace, istitile, partition, replace, join, split, count, decode, encode, swapcase, String constants, Regular Expressions and Pattern Matching

Lists: Concept of mutable lists, creating, initializing and accessing the elements, traversing, appending, updating and deleting elements, composition, lists as arguments

List operations: joining, slicing, + , * , in , not in

List functions and methods: len( ), insert( ), append( ), extend( ), sort( ), remove( ), reverse( ), pop( ), list( ), count( ), extend( ), index( ), cmp( ), max( ), min( )

Dictionaries: Concept of key-value pair, creating, initialising and accessing the elements in a dictionary, traversing, appending updating and deleting elements

Dictionary Functions and methods: cmp( ), len( ), clear( ), get( ), has_key( ), items( ), key( ), update( ), values( ), pop( ), fromkeys( ), dict( )

Tuples: Immutable concept, creating, initialising and accessing elements in a tuple, Tuple assignment, Tuple slices, Tuple indexing,

Tuple Functions: cmp(), len(), max(), min(), tuple(), index(), count(), sum(), any(), all(), sorted(), reversed()

Unit 3: 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, compilation , 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; Variables of built-in-datatypes, Declaration/Initialization of variables, Assignment statement, Type modifier: signed, unsigned, long

Operator and Expressions: Operators: Arithmetic operators (-,+,*,/,%),Assignment operator(=),C++ shorthands (+=,- =,*=,/=,%=) Unary operator (-), Increment(++) and Decrement (--) Operators, Relation operator (>,>=,<=,=,!=), Logical operators (!,&&,II),Conditional operator; Precedence of Operators; Automatic type conversionin expressions, Type casting;

Unit 4: Programming In C++

Flow of control:

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

Inbuilt Functions

- Standard input/output functions - stdio.h: gets ( ), puts ( )
- Character Functions - Ctype.h: isalnum ( ), isalpha ( ),isdigit ( ), islower ( ), isupper ( ), tolower ( ), toupper ( )
- String Function - string.h: strcpy ( ), strcat ( ), strlen ( ), strcmp ( ), strcmpi ( ), strev ( ),strlen ( ), strupur ( ), strlwr ( )
- Mathematical Functions - math.h: fabs ( ), pow ( ), sgrt ( ), sin ( ), cos ( ), abs ( )
- Other Functions - stdlib.h: randomize ( ), random ( )

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, scope rules; local and global variables.

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

Structured Data Type:

Arrays: Introduction to Array and its advantages.

One Dimensional Array: Declaration/initialization of One-dimensional array, inputting array elements, accessingarray 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/initialization of a two-dimensional array,inputting array elements accessing array elements, manipulationof 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, declaring structure variables,accessing structure elements, passing structure to functions as value and reference, function returning structure, array of structure

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