

CPP PROGRAMMING SYLLABUS



LEARN TODAY, LEAD TOMORROW!

Quality Training. Practical Learning. Real Results.

JOIN THE BEST COMPUTER CLASSES

✓ Expert Trainers ✓ Practical Learning ✓ Job Oriented Courses

WHY CHOOSE TEKZEN?



Industry
Experienced
Trainers



100% Practical
Hands-On
Learning



Live Projects &
Industrial
Training



100% Job
Assistance



Communication
Skills &
Personality Dev.

*Good Skills
Open All
Doors!*



UPGRADE
YOUR SKILLS
BUILD YOUR
FUTURE

 **TEKZEN**

 PRACTICAL LEARNING  LIVE PROJECTS  CERTIFICATE PROVIDED  PERSONALIZED ATTENTION  PLACEMENT ASSISTANCE



ENROLL NOW! →



LIMITED SEATS -
HURRY UP!



1st Floor Kalyani Market Near London Villas,

In Front of Shankeshwar City, Near Sri Aurobindo Hospital, **INDORE** |  **096858 25273**



info@tekzentechologies.com



www.tekzentechologies.com

Module 1: Introduction to C++ Programming

- What is C++? History, features, applications, and differences between C and C++
- Structure of a C++ program; main() function; compiling, linking, and executing
- Compiler/IDE setup; comments; namespaces and using declarations
- Character set, tokens, keywords, identifiers, constants, and variables
- Input/output using cin, cout, and basic stream manipulators
- Algorithms, flowcharts, pseudocode, and types of programming errors

Module 2: Data Types, Operators, and Expressions

- Fundamental data types, type modifiers, variables, constants, and initialization
- Type conversion, type casting, auto, and const
- Arithmetic, relational, logical, assignment, increment/decrement, bitwise, shift, and conditional operators
- Operator precedence and associativity
- sizeof operator, literals, and basic format/manipulation of output
- Practical programs using expressions and calculations

Module 3: Decision-Making and Loop Control

- if, if-else, nested if, else-if ladder, and switch-case
- while, do-while, and for loops; nested loops
- break and continue; introduction to goto and why it is generally avoided
- Menu-driven programs, loop tracing, and debugging
- Practical programs: even/odd, largest of numbers, prime, palindrome, Armstrong, perfect number, factorial, and Fibonacci
- Pattern printing and number-based problem-solving

Module 4: Functions and Recursion

- Function declaration, definition, prototype, function calls, and return values
- Parameters and arguments; call by value and reference
- Default arguments, inline functions, and function overloading
- Scope, local and global variables, and lifetime
- Recursion and standard library functions
- Modular programming and code reuse
- Practical programs: calculator, recursive factorial, Fibonacci, GCD, and prime checking

Module 5: Arrays, Searching, Sorting, and Strings

- One-dimensional and two-dimensional arrays; initialization and traversal
- Passing arrays to functions; insertion, deletion, and basic array operations
- Linear search, binary search, bubble sort, and selection sort
- C-style strings and C++ std::string
- String input/output, concatenation, comparison, searching, and common string methods
-

- Practical programs: maximum/minimum, reverse array, duplicate detection, matrix addition/transpose, palindrome string, and character frequency

Module 6: Pointers, References, and Dynamic Memory

- Pointers, addresses, dereferencing, and pointer initialization
- Pointer arithmetic and relationship between arrays and pointers
- References and passing arguments by reference
- Pointers with functions, pointer-to-pointer, and const pointers
- Dynamic memory allocation using new and delete
- Null, dangling, and uninitialized pointers; memory leaks and safe memory practices
- Practical programs: swap using pointers/references and dynamically allocated arrays

Module 7: Object-Oriented Programming and Classes

- Principles of OOP: abstraction, encapsulation, inheritance, and polymorphism
- Classes and objects; data members and member functions
- Access specifiers: private, public, and protected
- Data hiding, member functions, and this pointer
- Constructors, default constructors, parameterized constructors, and constructor overloading
- Destructors, object creation/destruction, and copy constructor basics
- Static data members, static member functions, const member functions, and friend functions
- Practical programs: Student, Rectangle, Employee, and Bank Account classes

Module 8: Inheritance and Polymorphism

- Introduction to inheritance and code reuse
- Single, multilevel, multiple, hierarchical, and hybrid inheritance
- Base and derived classes; access modes and constructor/destructor order
- Function overriding and the virtual function mechanism
- Compile-time and run-time polymorphism
- Virtual functions, dynamic binding, abstract classes, and pure virtual functions
- Object slicing and virtual destructors: introductory concepts
- Practical programs: Employee hierarchy and Shape area calculator

Module 9: Operator Overloading, Templates, and Exception Handling

- Operator overloading: rules, supported operators, and limitations
- Overloading unary and binary operators
- Type conversion basics and conversion constructors/operators
- Function templates and class templates
- Generic programming and reusable code
- Exception handling using try, catch, and throw
- Standard exceptions and basic error-handling practices
- Practical programs: Complex number operations, generic maximum/swap, and input validation

Module 10: File Handling

- Introduction to file handling and text/binary files
- File stream classes: ifstream, ofstream, and fstream
- Opening and closing files using open() and close()
- File opening modes: ios::in, ios::out, ios::app, ios::ate, ios::trunc, and ios::binary
- Reading and writing data using >>, <<, and getline()
- Appending and updating file contents
- File status and error handling
- Binary file operations using read() and write()
- Practical programs: file creation, reading, writing, appending, counting lines, and managing student records

MINI-PROJECTS

S. No.	Project Name	Key Features	C++ Concepts Used
1	Student Record Management System	Add, display, search, update, and delete student records	Classes, objects, functions, file handling
2	Banking Management System	Create accounts, deposit money, withdraw money, and check balance	Classes, objects, conditional statements
3	Library Management System	Add books, issue books, return books, and search books	OOP, arrays, file handling
4	Quiz Application	Display multiple-choice questions, calculate scores, and show results	Loops, arrays, functions, conditional statements
5	Number Guessing Game	Generate a random number and allow the user to guess it	Loops, random numbers, conditions
6	Employee Payroll System	Enter employee details, calculate salary, and generate salary slips	Classes, objects, functions
7	Inventory Management System	Add products, update stock, search items, and calculate bills	Arrays, classes, file handling
8	Electricity Bill Calculator	Calculate electricity bills according to unit consumption and tariff slabs	Functions, operators, if-else
9	Contact Management System	Add, search, edit, and delete contacts	Strings, structures/classes, file handling
10	ATM Simulation System	PIN verification, balance enquiry, cash withdrawal, and mini statement	Loops, functions, classes, conditions

Note: Prepare and solve weekly multiple-choice questions (MCQs) based on the topics covered during the week.