# FAIRFIELD

## INSTITUTE OF MANAGEMENT & TECHNOLOGY

**Affiliated to GGS IP University, an 'A+' Grade College by DHE, Govt. of NCT DELHI,
Approved by AICTE , Bar Council of India, NCTE and Recognised under 2(f) of UGC Act of 1956**

# CERTIFICATE COURSE

## ON

# PROGRAMMING IN

# C++

## APRIL 2025  –  JUNE 2025

### &

## SEPTEMBER 2025 – NOVEMBER 2025

## BLENDED  LEARNING / समिश्रित अधिगम

## ABOUT FAIRFIELD COLLEGE

The Fairfield Institute of Management and Technology (FIMT) is a NAAC accredited prestigious college in Delhi with an ambiance representing the academic culture of the capital town of the largest democracy in the world and a vibrant emerging economy. Established by the Fairfield Group of Institutions, it is an 'A' grade college approved by the Government of NCT of Delhi, affiliated with the prestigious Guru Gobind Singh Indraprastha University. FIMT is approved by All India Council of Technical Education (AICTE), Bar Council of India(BCI), National Council for Teacher Education (NCTE) and other authenticating bodies. Fairfield Group of Institutions is backed by a legacy of 50 years of shaping students' careers in the country in areas of Management, Commerce, Teachers' education, Journalism, Information and Communication Technology, Humanities, and Law & Legal Studies. The institutional distinction includes social outreach, field studies, and job orientation programs in respective areas of Study with specialized Add On Courses.

## INTRODUCTION OF THE CERTIFICATE COURSE

C++ is a general-purpose programming language that was developed as an enhancement of the C language to include object-oriented paradigm. It is imperative and compiled language. Students who have completed the basic C language find it easier to learn the C++ programming language. In fact, C++ is a high-level language designed for system and Bjarne Stroustrup at Bell Labs developed application programming that in 1983. C++ is an object-oriented, multi-paradigm language that supports procedural, functional, and generic programming styles. This course allows participants to experiment with design features through exercise programs. Starting with a presentation of C++ as a better C, the course explores function prototyping, the IOS Stream Library and operators, references, default arguments, structure and enum types, new keywords, comment styles, and free store operators. Once the features and syntax of the class construct in C++ are mastered, students are introduced to the methodology of object-oriented design and programming. The application of these concepts will in turn lead to more advanced uses of C++, including inheritance, parameterization (template classes), and dynamic binding. One of the key features of C++ is its ability to support low-level, system-level programming, making it suitable for developing operating systems, device drivers, and other system software. C++ has a large, active community of developers and users, and a wealth of resources and tools available for learning and using the language. C++ supports object-oriented programming, allowing developers to create classes and objects and to define methods and properties for these objects. C++ templates allow developers to write generic code that can work with any data type, making it easier to write reusable and flexible code. Standard Template Library STL provides a wide range of containers and algorithms for working with data, making it easier to write efficient and effective code. Overall, C++ is a powerful and versatile programming language that is widely used for a range of applications and is well suited for both low-level system programming and high-level application development.

## COURSE OBJECTIVES

- To understand how C++ improves C with object-oriented features.
- To learn how to write inline functions for efficiency and performance.
- To learn the syntax and semantics of the C++ programming language.
- To learn how to design C++ classes for code reuse.
- To learn how to implement copy constructors and class member functions.
- To understand the concept of data abstraction and encapsulation.
- To learn how to overload functions and operators in C++.
- To learn how containment and inheritance promote code reuse in C++.
- To learn how inheritance and virtual functions implement dynamic binding with polymorphism.
- To learn how to design and implement generic classes with C++ templates.
- To learn how to use exception handling in C++ programs.

## COURSE OUTCOME

- Demonstrate an understanding of algorithms in the problem-solving process.
- Identify the necessary properties of good problem-solving techniques.
- Create and analyze algorithms for solving simple problems.
- Use incremental program development to create, test, and debug algorithms for solving simple problems.

## EVALUATION METHODOLOGY OF THE CERTIFICATE COURSE

Evaluation methodology includes Hybrid Mode, Subjective and Objective Assessment, Presentation, class contribution and any other component as decided by the respective course faculties. A minimum of 75% attendance is a prerequisite for the successful completion of this program. The program may require participants to work on individual/group assignments and/or projects. The main objective of such assignments / projects will be to help the participants apply their conceptual learning in the program to actual organizational decision scenarios. The participants will have to secure the minimum passing marks in the respective evaluation. Participants who successfully complete the same and satisfy the requisite attendance criteria, will be awarded a certificate of completion. Participants who are unable to clear the evaluation criteria but have the requisite attendance will be awarded a participation certificate.

## PEDAGOGY

The pedagogy will be highly interactive in hybrid mode.It will consist of a judicious blend of lectures, real life case studies, quizzes & assignment.

## PROGRAMME CONTENT

- Introduction: Overview of OOPs concepts.
- C++ Programming basics: Variables, Expressions, Data types, Storage classes, Constants, Operators, Statements, Loops.
- Functions: Function Prototype, Function Definition, Function Calling, Scope Rules of Functions, Overview of Operator Functions, Inline Functions, Friend Functions Virtual Functions etc.
- Object and Classes: Defining Classes, Structures and Classes, understanding of classes and structures and union, Friend Functions, Friend classes, inline functions, Constructors and Destructors, Parameterized Constructors, Order of executions of constructors and destructors, Static class members i.e., Static data member and static member functions, Scope resolution operator, Nested Classes, Local Classes, Passing Object to Functions, Returning Object by Functions, Object Assignment etc.
- Arrays &Strings , Pointers and References : Arrays of object, Pointers to Object , this pointer pointers to derived type , pointer to class member , Reference parameter , Passing references, Returning references , References to derived type , C++ dynamic allocation operators, Strings.
- Function Overloading: Function Overloading, Overloading Constructors, Function Overloading and ambiguity, Overloading Binary and Unary Operators using Member Operator functions, Overloading Binary and Unary Operators using friend functions, Operator Overloading Restrictions, Overloading new and delete.
- Inheritance :Concept of inheritance, Derived class and based class , Base class access control Inheritance and protected members , Multiple Inheritance, Multilevel Inheritance , Constructors & Destructors in Inheritance ,Order of execution of constructors and Destructors Passing Parameter to base class constructors Virtual Base classes.
- Virtual Functions: Virtual Functions, calling virtual function through a base, pure virtual functions, abstract classes, using virtual functions, Early and Late binding.
- Streams and Files :C++ streams, C++ stream classes, C++ predefined streams, Formatted I/O, overloading the extraction and insertion operators, File classes, opening and closing file, reading & writing text files, unformatted and binary I/O, detecting EOF, Random access etc.
- Templates and Exceptions: Function templates, Class templates, Exceptions handling.

## HIGHLIGHTS OF THE CERTIFICATE COURSE

- This Certificate course holds significant merit for Teachers, seasoned In-Service Teachers, administrators and aspiring Professional Portfolio Pupil Teachers/Stuents.

- Duration of the Course: Three Months or 30 Hours.

- Mode: Hybrid (Offline/ Online).

- Medium of Instruction: Bilingual (Hindi/ English).

## ADMISSION PROCESS

The Online Session will commence from April 2025 to June 2025 & September 2025 to November 2025 (on Saturdays only.)The Admission Committee of FIMT shall make admission based on screening / merit in 10+2 or Bachelor's degree in any discipline followed by Group Discussion.

## ELIGIBILITY CRITERIA

An applicant who has successfully completed Senior Secondary School Course (10+2) or any equivalent with no age bar.

## CERTIFICATE

After Course Completion, Certificate will be provided by Guru Gobind Singh Indraprastha University(GGSIPU) New Delhi.

## REGISTRATION

- All interested candidates should register through the link **https://forms.gle/ZiLKSeWFL1a3sXzr7**
- Fee Structure:  10,000/- (₹)
- Examination fee : 1,000/- (₹)
- Payable through Online Mode.

Scan QR to pay

**FAIRFIELD INSTITUTE OF MANAGEMENT AND TECHNOLOGY**
DELHIC | TID : 64004068

| | |
|---|---|
| ACCOUNT HOLDER NAME | FAIRFIELD INSTITUTE OF MANAGEMENT & TECHNOLOGY |
| BANK NAME | HDFC BANK |
| ACCOUNT NUMBER | 50200024027620 |
| ACCOUNT TYPE | CURRENT |
| RTGS/NEFT/IFSC CODE | HDFC 0004404 |
| BRANCH | MAHIPALPUR |

# PROFESSIONAL COURSES @FIMT

**FIMT-SCHOOL OF LAW-** BBA-LL.B. INTEGRATED(H.) , BA-LL.B. INTEGRATED(H.) , LL.M.

**FIMT-SCHOOL OF ENGINEERING & TECHNOLOGY-** BCA , B.TECH. (CSE / IT / AI&ML)

**FIMT-SCHOOL OF JOURNALISM & MASS COMMUNICATION-** BA(JMC)

**FIMT-SCHOOL OF HUMANITIES & SOCIAL SCIENCES-.** BA-(H) ENG.

**FIMT-SCHOOL OF BUSINESS ADMINISTRATION-** BBA(GEN.)

**FIMT-SCHOOL OF COMMERCE-** B.COM.(H)

**FIMT-SCHOOL OF EDUCATION-** B.ED.