

DIPLOMA IN PROGRAMMING

C, C++ & Python

Duration: 6 Months | **Level:** Beginner to Intermediate | **Mode:** Theory + Practical + Projects
Practical Focus: 70% Practical / 30% Theory

Course Objective

A career-oriented programming diploma designed to take students from **zero programming knowledge to practical application development** using C, C++ and Python. The course develops programming logic, problem-solving, OOP, database skills, GUI development, API integration and project-building abilities.

PAGE 1 — PROGRAMMING FUNDAMENTALS & C

1. Programming Fundamentals

- Computer and programming concepts
- Programming languages and applications
- Compiler vs Interpreter
- IDE and code editors
- Program execution
- Problem-solving techniques
- Algorithms, flowcharts and pseudocode
- Debugging, testing and error identification
- Input → Processing → Output approach

2. C Programming

C Fundamentals

- Introduction, features and applications of C
- Program structure and compilation
- Constants, variables and data types
- printf() and scanf()
- Format specifiers and escape sequences
- Type conversion and casting

Operators & Control Statements

- Arithmetic, relational and logical operators
- Assignment, increment/decrement
- Conditional and bitwise operators
- Operator precedence and associativity
- if, if-else, nested if
- else-if ladder
- switch-case

Loops & Problem Solving

- for, while, do-while
- break and continue
- Number programs
- Pattern programs
- Prime, factorial, Fibonacci
- Armstrong, palindrome and perfect numbers

Functions

- Function declaration and definition
- Parameters and return values
- Local and global variables
- Call by value
- Recursion
- Library and user-defined functions

Arrays & Strings

- 1D and 2D arrays
- Array traversal
- Searching and sorting
- Character arrays and strings
- String functions and manipulation
- Matrix operations

Advanced C

- Pointers and pointer arithmetic
- Pointers with arrays, strings and functions
- Dynamic memory concepts
- Structures, unions and enumerations
- typedef
- Text and binary file handling
- Record management

C Projects

1. **Student Marks & Result Analyzer**
2. **Electricity Bill Generator**
3. **Employee Payroll System**

PAGE 2 — C++ & PYTHON FUNDAMENTALS

3. C++ Programming

C++ Fundamentals

- Introduction, features and applications
- C vs C++
- C++ program structure
- Namespaces
- cin and cout
- Variables, constants and data types
- Operators and type conversion
- Scope and references
- Conditional statements and loops
- Functions, recursion and default arguments
- Function overloading
- Inline functions

Object-Oriented Programming

- Procedural vs Object-Oriented Programming
- Classes and objects
- Encapsulation and abstraction
- Constructors and destructors
- Default, parameterized and copy constructors
- Constructor overloading
- Access specifiers
- Static members
- Friend functions/classes
- this pointer

Inheritance & Polymorphism

- Single inheritance
- Multilevel inheritance
- Multiple inheritance
- Hierarchical and hybrid inheritance
- Function overriding
- Operator overloading
- Virtual functions
- Pure virtual functions
- Abstract classes

Other C++ Topics

- Arrays and strings
- Pointers and references
- Dynamic memory concepts
- Exception handling
- try, catch, throw
- File streams
- Text and binary files
- Record management

C++ Projects

1. **ATM Simulation System**
2. **Restaurant Billing & Order System**
3. **Vehicle Rental Management System**

4. Python Programming

Python Fundamentals

- Python introduction, features and applications
- Python installation and environment
- IDLE, VS Code and Jupyter Notebook
- Python syntax and indentation
- Variables and dynamic typing
- Numbers, strings, Boolean and None
- Input/output and f-strings
- Operators
- if, elif, else
- for, while, range()
- break, continue, pass

Python Collections

- Strings and string methods
- Lists and nested lists
- List comprehension
- Tuples and packing/unpacking
- Sets and set operations
- Dictionaries and nested dictionaries
- Dictionary comprehension

Functions & Modules

- Functions and parameters
- Return values
- Default and keyword arguments
- *args and **kwargs
- Lambda functions

- Recursion
- Modules and packages
- pip
- Virtual environments
- Exception handling

Python OOP

- Classes and objects
- Attributes and methods
- self
- Constructors
- Instance and class variables
- Encapsulation
- Abstraction
- Inheritance
- Polymorphism
- Method overriding
- Special methods
- Properties
- Abstract classes

5. Python Files & Databases

File Handling

- Text files
- Reading, writing and appending
- File modes
- Directory management
- CSV files
- JSON files

Database Programming

- Database concepts
- DBMS vs RDBMS
- Tables, records and fields
- Primary keys and relationships
- CRUD operations
- SQL fundamentals
- Python database connectivity
- Insert, read, update and delete
- Parameterized queries
- Transactions and error handling

6. Python GUI Development with Tkinter

GUI Fundamentals

- GUI vs console applications
- Tkinter introduction
- Main window and properties
- Event-driven programming

Tkinter Widgets

- Label
- Button
- Entry
- Text
- Frame
- Checkbutton
- Radiobutton
- Listbox
- Combobox
- Spinbox
- Menu
- Canvas

GUI Development

- `pack()`, `grid()` and `place()`
- Event handling
- Callback functions
- Forms and validation
- Message boxes
- Login and registration forms
- GUI with files
- GUI with SQLite/database
- CRUD interfaces
- Search, update and delete interfaces

Tkinter Practice Applications

- Calculator
- Digital Clock
- Login Application
- Student Registration Form
- Expense Manager
- To-Do Application

7. Python API Integration

API Fundamentals

- What is an API?
- Client-server architecture
- Web services
- REST APIs
- Endpoints
- Requests and responses
- HTTP methods: GET, POST, PUT, PATCH, DELETE

API Development with Python

- requests library
- Query parameters
- Headers
- JSON requests and responses
- HTTP status codes
- JSON parsing
- API authentication
- API keys
- Environment variables
- API error handling
- Timeouts and connection errors
- API usage limits

API-Based Applications

- **Weather Information App**
 - **Currency Converter**
 - **Random Quote Generator**
 - **News Headlines Viewer**
 - **Public Data Explorer**
-

8. Python Automation & Data Analysis

Automation

- File and folder automation
- File organization and renaming
- CSV/JSON processing
- Report generation
- Date/time automation
- Repetitive task automation

Data Analysis Foundation

NumPy

- Arrays
- Array operations
- Mathematical calculations
- Basic statistics

Pandas

- Series and DataFrames
- CSV/Excel import
- Filtering and sorting
- Grouping
- Missing data
- Aggregation

Matplotlib

- Basic charts
- Bar, line and pie charts
- Simple data visualization

9. Git, GitHub & Professional Skills

Version Control

- Git fundamentals
- Repository creation
- `git init`, `add`, `commit`, `status`, `log`
- Branching and merging

GitHub

- Repository creation
- Project uploading
- README files
- Portfolio development

Professional Programming

- Clean coding practices
- Modular programming
- Code documentation
- Debugging and testing
- Test cases and edge cases
- Basic code optimization
- Programming problem solving

10. PROJECT & ASSESSMENT

Major Projects

C: Employee Payroll System

C++: Vehicle Rental Management System

Python: Personal Expense Tracker

Students also complete **100+ programming exercises, logic-building problems, GUI exercises and API exercises** throughout the course.

LEARNING OUTCOMES

After completing the diploma, students will be able to:

- ✓ Program confidently in **C, C++ and Python**
- ✓ Develop strong programming and problem-solving logic
- ✓ Apply Object-Oriented Programming
- ✓ Work with files and databases
- ✓ Build desktop applications using **Tkinter**
- ✓ Integrate **REST APIs using Python**
- ✓ Process JSON, CSV and Excel data
- ✓ Perform basic data analysis with Python
- ✓ Use Git and GitHub
- ✓ Build and present real-world projects
- ✓ Create a professional programming portfolio
- ✓ Prepare for internships and entry-level coding careers

COURSE ROADMAP

Programming Fundamentals → C → C Projects → C++ → OOP → C++ Projects → Python → OOP → Files & Database → Tkinter → API Integration → Automation → Data Analysis → Git/GitHub → Projects → Portfolio & Interview Preparation