

ELECTIVE III - C
(Theory & Practical)

Subject Title: PROGRAMMING IN C++ *

No. of Hours: 3

Subject Description: This paper presents the importance of class structure, operators, the types of inheritance and polymorphism, file handling.

Goals: To enable the students to learn about the class structure, operators, inheritance, polymorphism, file handling.

Objectives: On successful completion of the course the students should have learnt class structure, member functions & data members.
Learnt the concept of inheritance, types and example problems.
Learnt the concepts of polymorphism, types and problems.
Learnt the concepts of File handling.

UNIT-I:

Evolution of C++ - applications of C++ - structure of C++ program. Tokens – keywords – identifiers and constants – basic data types – user-defined data types – constant pointers and pointers to constants – symbolic constants –type compatibility – declaration of variables – dynamic initialization of variables – reference variables – operators in C++ - scope resolution operator – memory management operators – manipulators – type cast operator – expressions and their types – special assignment expressions – implicit conversions – operator precedence.

UNIT-II:

Functions in C++ : The main function – function prototyping – call by reference – return by reference – inline functions – default arguments – const arguments – function overloading.
Managing Console I/O Operations: C++ streams – C++ stream classes – unformatted console I/O operations – formatted console I/O operations –managing output with manipulators.

UNIT-III:

Classes and Objects: Specifying a class – defining member functions – making an outside function inline – nesting of member functions – private member functions – arrays within a class – memory allocation for objects –arrays of objects – objects as function arguments – friend functions – returning objects – const member functions.
Constructors and Destructors: Introduction – constructors – parameterized constructors – multiple constructors in a class – constructors with default arguments – copy constructor.

UNIT-IV:

Operator Overloading: Introduction – defining operator overloading – overloading unary operators – overloading binary operators - overloading binary operators using friends – rules for overloading operators.
Inheritance: Introduction – defining derived classes – single inheritance – making a private member inheritable – multilevel inheritance – multiple inheritance – hierarchical inheritance – hybrid inheritance.

UNIT-V:

Working with Files: Introduction – Classes for File Stream Operations - Opening and Closing a File – Detecting End-of-file – More about open(): File Modes – File Pointers and their Manipulations – Sequential Input and Output Operations – Updating a File: Random Access.

Text Books:

1. E.Balagurusamy - 'Object Oriented programming with C++', McGraw Hill.
2. Robert Lafore – 'Object oriented programming in Turbo C++', Galgotia publications Pvt.Ltd, New Delhi- 110002 11994.
3. Bjarne Stroustrup – 'The C++ programming language', II Edition, Addison Wesley, 1991.

Reference Books:

1. D.Ravi Chandran – 'Programming with C++', Tata McGraw-Hill publishing company limited (1996), New Delhi.
2. Ashok N.Kamthane – 'Object Oriented Programming with ANSI and Turbo C++', Pearson Education publishers (2003).
3. John R.Hubbard – 'Programming with C++', 2nd Edition, TMH publishers (2002).

PROGRAMMING IN C++ PRACTICAL LIST.

1. Write a function 'power()' to raise a number 'm' to a power 'n'. The function takes a 'double' value for 'm' and 'int' value for 'n', and returns the result correctly. Use a default value of 2 for 'n' to make the function to calculate squares when this argument is omitted. Write a 'main()' that gets the values of 'm' and 'n' from the user to test the function.
2. Write a program to compute compound interest of a given amount AMT for 'n' years. Use function overloading so that the program gets input of interest rate RATE in any of the data type 'float' or 'int'.
3. Create a class which consist of employee detail ENO, ENAME, DEPT, BASIC SALARY. Write a member function to get and display them. Derive a class PAY from the above class and write a member function to calculate DA, HRA and PF depending on the grade and display the payslip in a neat format using console I/O.
4. Define two classes POLAR and RECTANGLE to represent points in the polar and rectangle system. Write a program to convert from one system to another.
5. Create a class FLOAT that contains one float data member. Overload all the four arithmetic operators so that they operate on the objects of FLOAT.