Programming Fundamentals using C++ (BHCS01) Discipline Specific Core Course - (DSC)

Credit: 06

Course Objective

This course is designed to develop structured as well as object-oriented programming skills using C++ programming language. The course not only focuses on basic C++ constructs but also covers object-oriented programming features in-depth, namely Encapsulation, Abstraction, Inheritance and Polymorphism for writing efficient codes.

Course Learning Outcomes

On successful completion of the course, students will be able to:

1. Explain significance of object oriented paradigm

2. Solve programming problems using object oriented features.

3. Handle external files as well as exceptions.

4. Reuse classes to create new classes.

5. Handle exceptions in programs.

Detailed Syllabus

Unit 1

Introduction to C++: Overview of Procedural Programming and Object-Oriented Programming, Using main () function, Header Files, Compiling and Executing Simple Programs in C++.

Unit 2

Programming Fundamentals: Data types, Variables, Operators, Expressions, Arrays, Keywords, Naming Convention, Decision making constructs (if, switch), Looping (for, while, do…while), Type Casting, Input-output statements, Functions, Command Line Arguments/Parameters.

Unit 3

Object Oriented Programming: Overview of Abstraction, Encapsulation, Inheritance, and Polymorphism. Creating Classes and objects, Modifiers and Access Control, Constructors, Implementation of Inheritance (Single and multilevel), Implementation of Polymorphism (Function Overloading and Operator Overloading, Function Overriding).

Unit 4

Pointers and References: Static and dynamic memory allocation, Pointer and Reference Variables, Pointers vs. References, Implementing Runtime polymorphism using pointers and references.

Unit 5

Exception and File Handling: Using try, catch, throw, throws and finally; Nested try, creating user defined exceptions, File I/O Basics, File Operations.

Practical

1. Write a program to compute the sum of the first n terms of the following series:
$$S = 1 - 1/(2^2) + 1/(3^3) - \ldots 1/(n^n)$$

   where $^$ is exponentiation.
   The number of terms n is to be taken from user through command line. If command line

   argument is not found then prompt the user to enter the value of n.

2. Write a program to remove the duplicates from an array.

3. Write a program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.

4. Write a menu driven program to perform following operations on strings (without using inbuilt string functions):

   a) Show address of each character in string

   b) Concatenate two strings.

   c) Compare two strings

   d) Calculate length of the string (use pointers)

   e) Convert all lowercase characters to uppercase

   f) Reverse the string

5. Write a program to merge two ordered arrays to get a single ordered array.

6. Write a program to search a given element in a set of N numbers using Binary search

   (i) with recursion (ii) without recursion.

7. Write a program to calculate GCD of two numbers (i) with recursion (ii) without recursion.

8. Create Matrix class. Write a menu-driven program to perform following Matrix operations:

   a) Sum

   b) Product

   c) Transpose

9. Define a class Person having name as a data member. Inherit two classes Student and Employee from Person. Student has additional attributes as course, marks and year and Employee has department and salary. Write display() method in all the three classes to display the corresponding attributes. Provide the necessary methods to show runtime polymorphism.

10. Create a class Triangle. Include overloaded functions for calculating area. Overload assignment operator and equality operator.

11. Write a program to read two numbers p and q. If q is 0 then throw an exception else display the result of p/q.

12. Rewrite Matrix class of Q8 with exception handling. Exceptions should be thrown by the functions if matrices passed to them are incompatible and handled by main() function.

13. Create a class Student containing fields for Roll No., Name, Class, Year and Total Marks. Write a program to store 5 objects of Student class in a file. Retrieve these records from file and display them.

14. Copy the contents of one text file to another file, after removing all whitespaces.

References

1. Forouzan & Gilbert (2012). *Computer Science: A Structured Approach Using C++*. Cengage Learning.

2. Schildt, H. (2003). *C++: The Complete Reference*. 4th edition. Tata McGraw-Hill.

Additional Resources

1. Balaguruswamy, E. (2017). *Object Oriented Programming with C++* (7th ed.). McGraw Hill Education.

2. Kanetkar, Y. P. (2015). *Let us C++* .2nd edition. BPB Publishers.

3. Prata, S. (2015). *C++ Primer Plus* 6th edition. Pearson Education India.

4. Stroustrup, B. (2013). *The C++ Programming Language* .4th Edition. Pearson Education.