

GUJARAT TECHNOLOGICAL UNIVERSITY

Master of Computer Application

Subject Name : Object Oriented Concepts and Programming (OOCp)

Subject Code : 620003

Objectives:

The programming for small devices like mobile phones, networking devices like routers, coding for graphics and multimedia, requires efficient coding as well as object oriented programming. The C++ language fits perfectly as a tool for this type of work. How this important language is to be mastered and how to use this knowledge in building efficient and flexible code is one of the prime requirements today. The course presented here is targeting to enable the student to master such skills. Aim of the course is to enable students to

1. Differentiate between procedural and object oriented programming.
2. Learn C++ as a language and various features of it.
3. Learn Object Oriented principles and their application using C++.

Prerequisites:

1. Knowledge of C language
2. Programming concepts including algorithm building and logic

Contents :

1. Introduction to C++, Overview of Core C++ Language, Classes and Objects

Identifiers and constants (Literals), Keywords, Data Types, The Operators, New Casting Operators, typeid and throw, The Conditional structures and Looping Constructs, , The Difference between struct and class in C++, The difference between Union and Class, Static Data members of a class, Pointer to objects and pointer to members of class, The local classes, Assigning Objects

2. Functions

Introduction, The inline function, Default Arguments to the function, Functions with object as parameters, Call by reference and return by reference, Prototyping and Overloading, Friend functions, Const and Volatile functions, Static functions, Private and Public functions, Function Pointers, Adding C functions to the C++ program

3. Constructors and Destructors

Introduction to constructors, The explicit constructors, Parameterized constructors, Having multiple constructors, Constructors with default arguments, Dynamic Initialization, Constructor with dynamic allocation, copy constructors, The member initialization list, destructors

4. Operator Overloading and User Defined Conversions

Introduction, Unary Operators, Binary Operators, Using Friends as operator functions, Overloading other Operators, The need for user defined conversion, Four different cases where user defined conversions are needed, Comparison of both the methods of conversion

5. Templates

Function Templates, Non Generic (Non Type) Parameters in Template functions, Template function and specialization, Overloading a template function, Using Default Arguments, Class Templates, Classes with multiple generic data types, Static data members, Primary and Partial Specialization, The Export Keyword, The other use of typename

6. Inheritance

The need, Defining derived class using single base class, Derivation using public, private and protected access modifiers, The implementation of inheritance in the C++ object model, The Access Control, The Access Declaration, The multiple-inheritance, Abstract classes, Composite objects (container objects)

7. Runtime polymorphism and Run Time Type Identification

Compile Time and Runtime Polymorphism, Pointers to Objects, This pointer, Compatibility of Derived and base class pointers, The subobject concept, Virtual functions, Static invocation of virtual function, Default arguments to virtual functions, Virtual destructors, Pure virtual functions, Run Time Type Identification

8. Exception Handling

Introduction, Need for Exception handling, Components of exception handling mechanism, rethrowing an exception , Terminate and Unexpected functions, Drawbacks of exception handling approach, The exception Class

9. IO Streams

Need for streams, Advantages of using C++ I/O over C IO, The C++ Predefined streams, Formatting IO, Formatting using ios members, Manipulators, Creating our own manipulator

10. Using Files for IO

Why IO is special, Text and binary streams, Opening and closing files, Dealing with text files

Dealing with binary files, Providing Random Access using seek, IO Modes, Handling Errors

11. Namespaces

Introduction and need, Use the using syntax, Defining namespaces, Extending the namespace, Unnamed namespaces, Nested Namespaces, Namespace aliases, The std namespace, The Koenig lookup, Overhead with namespaces

12. The Standard Template Library

The STL (Standard Template Library) Introduction, Generic Programming , Generic Software Components, Generic Algorithms, Iterators, Containers, Algorithms

Main Reference Book:

1. Programming with ANSI C++ by Bhushan Trivedi, Oxford University Press

Suggested Additional Reading:

1. Object Oriented programming with C++ by E. Balagurusamy, TMH
2. Complete Reference C++ by Herbert Schildt McGraw Hill Publications
3. Computer Science- A Structured approach using C++ by Forouzan ,Gilburg, THOMSON Books
4. Object Oriented programming in C++ by Robert Lafore, Pearson Education
5. C++ FAQs by Pearson Education
6. C++ Primer by Stanley Lippmann, Pearson Education
7. The C++ Programming Language by Bjarne Stroustrup, Pearson Education
8. Effective C++ by Scott Mayer Addison Wesley

Chapter-wise coverage from main reference book :

Chapters : 1,2,3,4,5,6,7,8,9,10,11,12,13,14,16.

Accomplishments of the student after completing the course :

1. He/She should be able to understand and appreciate the Object Oriented approach of programming
2. He/She should be aware of the working and architectural model of C++.
3. He/She should be able to solve problems given to him/her using C++ with keeping balance between efficiency and flexibility