

PRG300 : C++ Programming

Code : PRG300

Duration : 5 days

Category : Programming

Audience :

This course is targeted at developers who want to learn programming in C++ and others who want to understand C++ code.

Prerequisites :

Knowledge of and experience with C programming is required to attend this course.

Realization :

The theory is treated on the basis of presentation slides and is interspersed with exercises. Illustrative demos are used to clarify the discussed concepts. The course material is in English.



C++ Programming



Contents :

In the course C++ Programming participants learn how to program in the C++ language. First the differences between C and C++ are discussed concerning variable declarations, formatted output with the stream IO library, namespaces, function overloading and default function parameters. Subsequently the new C++ reference variables are discussed. An important element of the course is the C++ class concept and C++ implementation of object-oriented principles such as abstraction and encapsulation. Attention is paid to dynamic memory allocation with new and delete and the role of assignment operators and copy constructors. Also special features of classes such as statics, friends and iterators are discussed. Next the object-oriented principles of inheritance and polymorphism are part of the subject matter. This includes the concepts of virtual functions, v-tables, dynamic binding and abstract classes. C++ has the option to give existing operators a different meaning and this phenomenon is discussed in the module operator overloading. Attention is paid to important features of the standard C++ library like the String class and the base concepts of C++ templates and the Standard Template Library (STL). Finally exception handling and how this is implemented in C++ is addressed.

Module 1 : Intro C++

- C++ Features
- Comments in C++
- I/O Stream Library
- Output and Error Stream
- Input Stream
- Multiple Input Values
- IO Stream Function get
- Formatted Output
- Function Prototypes
- Default Function Arguments
- Inline Functions
- Macro Versus Inline Function
- Overloading Functions
- Variable Declaration
- Namespaces
- Scope Resolution Operator

Module 2 : References

- References
- Reference Initialization
- References and Pointers
- Reference to Constant
- Passing References
- Comparison Parameter Passing
- References as Return Values
- Potential Error
- Potential Problem
- Returning Reference to Global

Module 3 : Classes

- Classes in C++
- New Data Type in C
- Class Declaration
- Formal Class Declaration Syntax
- Class Sections
- Constructor
- Destructor
- Class Implementation
- Member Functions
- Class Usage
- Calling Member Functions
- Reading Data Members
- Setting Data Members
- Header and Sources Files
- Advantages Access Functions
- References to private Data
- Constant Objects
- Member Objects
- Composition Example

Module 4 : Dynamic Memory Allocation

- Dynamic Memory Allocation
- new Operator
- delete Operator
- Dynamic Arrays
- Classes with Pointer Data
- Assignment Operator
- Self-Assignment Problem
- this Pointer
- Using this Pointer
- Chained Assignments
- Assignment and Initialization
- Initialization with Copy Constructor
- Copy Constructors
- Passing Objects
- Returning Objects
- Passing References to Objects

Module 5 : Class Features

- Static Members
- Accessing Static Data Members
- Initialization of Static Data Members
- Static Member Functions
- Friends
- Friend Functions
- Iterator Class
- Arrays of Objects
- Initializing Object Arrays
- Free Store and Class Arrays
- _set_new_handler Function
- Overloading new and delete
- Class Specific new and delete

Module 6 : Inheritance and Polymorphism

- Inheritance
- Type Relations in C and C++
- Derived Classes in C++
- Class Hierarchy
- Redefining Member Functions
- Calling Redefined Members
- Derived Class Constructors
- Base – Derived Class Conversion
- Pointer Conversions
- Virtual Functions
- Polymorphism
- Dynamic Binding
- Virtual Function Implementation
- Virtual Function Table
- Pure Virtual Functions
- Abstract Classes
- Multiple Inheritance
- Duplication of Data Members
- Virtual Derivation

Module 7 : Operator Overloading

- Operator Overloading
- Overloading for Numeric Types
- Overloading Rules
- Overloading Restrictions
- Not Overloadable Operators
- When not to Overload
- Numeric Class Overloading
- Operators as Friend
- Unary Overloading Operator

Module 8 : Templates

- Function Templates
- Macro's versus Templates
- Using Template Functions
- Template Function Overriding
- Class Templates
- Template Array Class
- Template Class Definition
- Template Class Instantiation
- Output Operator as Friend
- Template Class Specialization
- Template Class Static Members
- new and delete in Template Classes
- Template Initialization Statics
- Memory Allocation Statics
- Standard Template Library (STL)

Module 9 : Exception Handling

- Exception Handling in C++
- Memory Exhaustion Handling
- Exception Handling Principles
- Throwing Exceptions
- try Block
- catch Handlers
- Division By Zero Example
- Benefits Exception Handling
- Multiple catch Handlers
- Template Array Class
- Exceptions Array Class
- Matching catch Order
- throw List