Programming in C++11 : Basic / Intermediate

COURSE DESCRIPTION

This **course** will **present** the **syntax** and **constructs** of the **ISO Standard C++11** programming language. Basic, intermediate, and some advanced techniques will be shown. **All features shown are applicable to all C++11 variants (on Windows, Unix, and Linux systems).**

COURSE OBJECTIVES

The overall **course objective** is to **present** sufficient **C++11** language **information** to have the **programmer** at **production coding level** by the **end** of the training **course**.

COURSE TOPICS

Overview of object oriented programming
      evolution of programming types
      encapsulation
      polymorphism
      inheritance
      modularity and abstraction

Basic I/O (Operations) in C++
      istream class
      ostream class
      cout and **cin I/O** object (functions)
      I/O manipulators

Namespaces
      the Global Name Space (**GNS**)
      the **std** namespace
      creating namespaces
      scoping identifiers in namespaces

Programming in C++11 : Basic / Intermediate

COURSE TOPICS

Defining and using classes
  basic class definition
  private and public members
  instantiation of (class) objects
  constructors and destructors
  dynamic memory allocation/deallocation
  arguments to constructors
  defining and using template (generic) classes
  using **RTTI** to determine object types

Arrays, pointers, and references in C++11
  object pointers
  the **this** pointer
  arrays of objects
  smart pointers
  references

Exception Handling
  overview of **C++11** exception handling
  establishing a **try** block with **catch** handlers
  different methods of **throw**ing an exception

Defining and using Functions in C++11
  class member functions
  overloading function capabilities
  passing objects to functions
        by value (copy)
        by reference
  overloading constructor functions
  template functions
  default arguments
  operator overload(ing functions)

Programming in C++11 : Basic / Intermediate

COURSE TOPICS

Inheritance
  base and derived classes
  access control
  constructors and destructors
  multiple inheritance
  virtual base classes
  virtual functions

The Standard Template Library
  history of the library
  containers, algorithms, references, and iterators
  using the **vector**, **map**, and **string** templates

COURSE DURATION

This course normally requires **three** (3) days, approximately 50 % lecture and 50 % programming lab time.

COURSE PREREQUISITES

It is **assumed** that the **participant** has **production** level **experience** and **knowledge** of the **ISO / ANSI Standard C** programming language.