

# PRG303 : Advanced C++ Programming

Code : PRG303 Duration : 3 days

## Audience :

This course is intended for developers who are familiar with C++ but who wish to delve into the more advanced techniques and features of the language.

## Prerequisites :

To join this course knowledge of the basic concepts of C++ and extensive programming experience in C++ is required. Students should be familiar with topics like as operator overloading, templates, virtual functions and polymorphism.

## Realization :

The concepts are treated on the basis of presentation slides. Illustrative demo's are used to clarify the treated concepts. Theory and practice are interspersed and there is ample opportunity to practice.

## Contents :

In the course Advanced C++ Programming advanced concepts of the C++ language are discussed. The general structure of the Standard Template Library (STL) as well as its use in practice is addressed. Attention is paid to the details of containers, iterators and algorithms. Also advanced STL features such as function objects, binders, comparators and adapters are part of the subject matter. The course also deals with more advanced features of C++ such as the overloading of operators like  $\rightarrow$ ,  $\rightarrow^*$  and  $()$ , member templates and pointers to member functions. In addition a detailed look is taken at the inner workings of smart pointers, a technique that makes it possible to create objects somewhere without having to clean them up explicitly, like with Java garbage collection. The combination of smart pointers with STL and using hash maps and function objects is also discussed when treating advanced aspects of STL.

### Module 1 : Templates Review

- Templates review
- Template specializations and partial specializations
- Non-type template parameters
- Default template parameters
- Template member functions

### Module 4 : STL Containers

- Vectors
- Maps
- Hash Maps
- Strings Bitsets
- Function objects
- Predicates, and comparators

### Module 7 : Smart Pointers

- Smart Pointers
- Smart Pointers and STL

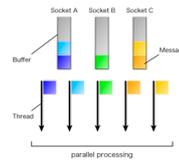
### Module 2 : STL Intro

- Introduction to the STL
- Purpose and structure of the STL
- Containers
- Algorithms
- Iterators
- Container classification
- Characteristics and content

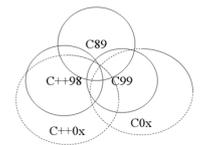
### Module 5 : Adapters

- Container adapters
- Iterator adapters
- Function adapters
- Iterator traits and tags
- Adapting existing code to STL

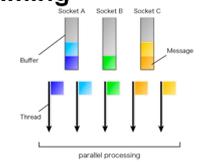
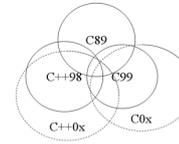
## Category :



## Programming



## Advanced C++ Programming



### Module 3 : Iterators and Algorithms

- Iterator classification and behavior
- Generic algorithms and iterators
- Design of generic algorithms
- Reverse iterators
- iostream iterators

### Module 6 : Operator Overloading

- Advanced Operator Overloading
- Functors Or Function Objects
- Pointers To Member Functions
- Member Templates
- Reference Counting