

Programming in C++11 for C Programmers : Part 2

COURSE DESCRIPTION

This **course** will **present** the **additional capabilities** of the **ISO Standard C++11** programming language. **Intermediate** and **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 **provide additional C++11 language features** in each of the **coverage areas** from **Part 1**.

COURSE TOPICS

Namespaces

- namespace issues
- collisions
- pollutions
- namespace aliases
- inlined namespaces

Advanced I/O (Operations) in C++

- I/O extractors
- I/O inserters
- object oriented file I/O

Defining and using classes

- constructors
 - default** keyword
 - delete** keyword
- template classes
 - template aliases
 - variadic templates

Programming in C++11 for C Programmers : Part 2

COURSE TOPICS

Pointers and references in C++11

smart pointers

unique_ptr

shared_ptr

weak_ptr

references

move semantics

move reference (operations)

move constructor

Exception Handling

using **noexcept**

using **std::set_terminate()**

Data Operations in C++11

const_cast

static_cast

brace initialization

passing objects to functions

by value (copy)

by reference

by move reference

Inheritance

using **override**

using **final**

Functions

lambdas (anonymous functions)

closures

captures

trailing return type

Programming in C++11 for C Programmers : Part 2

COURSE TOPICS

The Standard Template Library
begin() and **end()** methods
tuple container
regular expressions
thread operations
random number operations
clock and timing capabilities

Miscellaneous Features

type aliases
decltype
type traits
static assert

COURSE DURATION

This course normally requires **two** (2) days, approximately 50 % lecture and 50 % programming lab time.

COURSE PREREQUISITES

It is **assumed** that the **participant** has **attended** the **Programming in C++11 for C Programmers: Part 1** course, or has **production** level **coding experience** with (any version standard) of **C++**.