

Programming in C++ (for C Programmers)

© 1998, 2001, 2005, 2009 Rex Jaeschke. All rights reserved.

COURSE OVERVIEW:

This course teaches students how to program in Standard C++. A significant lab component is included. The course is not hardware or operating system-specific.

COURSE LENGTH: 4–5 days.

GOALS:

Provided students meet the prerequisites, at the end of the course, they should:

- Have a working knowledge of the language's statements and operators.
- Be conversant about the standard runtime library.
- Have been exposed to issues regarding the importance of programming style.
- Be aware of the strengths and weaknesses of C++ and what it will take to master and exploit it.
- Be able to design and implement simple classes.
- Have a basic understanding of encapsulation, inheritance, and polymorphism.
- Understand templates.
- Understand exception handling.

WHO SHOULD ATTEND:

Programmers and technical managers who are seriously interested in, or are about to begin, programming in the C++ language; or who wish to evaluate the suitability of C++ for projects and/or programming personnel.

PREREQUISITES:

A thorough working knowledge of C is assumed. As C++ is essentially a superset of C, the more you know about C, the easier will be the transition to C++, at least from a syntactic viewpoint. Unless you are very comfortable with the following topics, you may well have trouble keeping up with the C++-specific theory and making adequate progress with the lab problems:

- All aspects of structures.
- All aspects of enumerated types.

- Purpose and syntax of data pointers.
- The `const` type qualifier.
- Function prototypes and new-style function definitions.
- Passing arguments by value versus address.
- The use of functions versus macros with arguments.
- Dynamic memory allocation via `malloc` and friends.
- Have a feel for C's strengths and weaknesses.

MATERIALS:

Each student will receive the following materials:

- *Programming in C++* — This textbook was written specifically for teaching Standard C++. It contains all of the main features added during the standardization of the language. This book serves as a useful reference once the course has been completed.

DETAILED TOPICS:

The main topics covered are:

- The new types `bool` and `string`
- Function overloading
- Function inlining
- Function templates
- References
- Default argument values
- Dynamic memory allocation
- I/O
- Introduction to classes
- Constructors and destructors
- Operator overloading
- Inheritance
- Exception handling
- Class templates