View Full Course Details including Latest Schedule Online

PHOENIX TS

C++ Programming Training

This course will give you a new programming skill to add to your resume by learning C++.

BONUS! Cyber Phoenix Subscription Included: All Phoenix TS students receive complimentary ninety (90) day access to the Cyber Phoenix learning platform, which hosts hundreds of expert asynchronous training courses in Cybersecurity, IT, Soft Skills, and Management and more!

Course Overview

Our 5-day, instructor-led C++ Programming Training course is designed to teach non-programmers the following:

- About the core C features of C++
- C++ class construct
- About memory management, scope, and access control mechanisms
- About Polymorphisms and virtual functions and inheritance

No previous programming experience is necessary.

Schedule

Currently, there are no public classes scheduled. Please contact a Phoenix TS Training Consultant to discuss hosting a private class at 301-258-8200.

Course Outline

Classes

- Creating a data structure
- Methods
- Object scope
- C++ input and output
- Namespaces
- Data abstraction
- Enforcing data encapsulation
- File organization
- Classes in C++
- Objects
- This pointer

Constructors and Destructors

- Debug output
- The default constructor
- When are constructors called?
- The destructor
- The copy constructor
- Other constructors
- Why did it work before?
- Composition
- The report class
- Code reuse
- Initialization lists
- Labs

Inheritance

- Inheritance
- Bugreport
- Protected access modifier
- Access and inheritance
- Constructors and inheritance
- Initialization lists revisited
- Multiple inheritance
- Labs

Virtual Functions

- Inheritance and assignment
- Inside report's assignment operator
- Using pointers – a quick look at basics
- Class assignment and pointers
- Class assignment and pointers – a picture
- Static binding
- Polymorphisim
- The show_rep() function
- Using the show_rep() function
- Designing member function inheritance
- Labs

Pure Virtual Functions

- Bugfix and its relationship with bugreport
- Bugfix: association with bugreport
- Using bugfix with show_rep()
- Adding bugfix to the hierarchy
- Coding for the document class
- Reexamining the document class
- Pure virtual functions
- Designing member function inheritance
- Labs

References and Constants

- References
- Displaying references
- Changing references
- Pass by references
- Returning by reference
- Constant variables
- Constant references
- Constant methods
- Labs

New And Delete

- New and delete
- Array allocation
- The report class
- Compiler version of the copy constructor
- Guidelines for copy constructors
- The report constructors and new
- The report destructor and delete
- Virtual destructors
- Labs

Casting in C++

- Casting: a review
- New casting syntax
- Creating a string class
- The string class
- The conversion constructor
- Expanding our casting options
- Casting operator
- Using the casting operator
- Labs

Class Methods and Data

- Class data
- Class methods
- Using the new data
- More on class methods
- Labs

Overloaded Functions

- Function overloading
- Using overloaded functions
- Rules for overloading
- Overloading based on constness
- Default arguments

- Invoking functions with default arguments
- Labs

Overloaded Operators

- The basics of overloading
- Overloading with operator =+
- Coping with community
- Non-cummutative operators
- Friends and their problems
- The assignment operator
- Overloading the << Operator
- Using date with cout
- Labs

Exception Handling

- Why exception handling?
- try/catch/throw
- Exception classes
- Standard exception hierarchy
- Multiple catch blocks
- Catching everything
- Unhandled exceptions
- Exception in constructors and destructors
- Designing for exceptions
- Labs

Standard Template Library

- Class template concepts
- Standard template library (STL) overview
- Containers
- Iterators
- Iterator syntax
- Non-mutating sequential algorithms
- mutating sequential algorithms
- Sorting algorithms
- Numeric algorithms
- auto_ptr class
- string class

- Labs

STL Containers

- Container classes
- Container class algorithms
- vector class
- Additional vector class methods
- deque class
- list class
- set and multiset classes
- map and multimap classes
- Labs

C++ Programming Training FAQs

Who should enroll in this course?

Professionals who benefit from C++ programming skills or individuals looking to being a programming career.

BONUS! Cyber Phoenix Subscription Included: All Phoenix TS students receive complimentary ninety (90) day access to the Cyber Phoenix learning platform, which hosts hundreds of expert asynchronous training courses in Cybersecurity, IT, Soft Skills, and Management and more!

Phoenix TS is registered with the National Association of State Boards of Accountancy (NASBA) as a sponsor of continuing professional education on the National Registry of CPE Sponsors. State boards of accountancy have final authority on the acceptance of individual courses for CPE credit. Complaints re-garding registered sponsors may be submitted to the National Registry of CPE Sponsors through its web site: www.nasbaregistry.org

Starting at **$2,190**

ATTENTION
For GSA pricing or Contractor quotes call
301-258-8200 – Option 2.

GSA

Price Match Guarantee

We'll match any competitor's price quote. Call us at 240-667-7757.

This **C++ Programming Training** course includes:

- 5 days of instructor-led training
- C++ Programming Training book
- Notepad, pen and highlighter
- Variety of bagels, fruits, doughnuts and cereal available at the start of class*
- Tea, coffee and soda available throughout the day*

- ○ Freshly baked cookies every afternoon*

*denotes this benefit is only available at participating locations