Introduction to Programming Concepts

Duration: Approximately 7 hours of coursework, to be completed in a four week span.

Price: This course is free when taken with any other HOTT course.

Delivery Options: This course is available via **on-demand** training.

Students Will Learn

- Basic computer terminology and jargon
- Programming tasks: Analysis and requirements gathering
- Programming tasks: Design and planning
- Programming tasks: Coding
- Programming tasks: Testing
- Programming tasks: Deployment/Documentation
- Programming tasks: Maintenance
- The nature, pros and cons of compiled vs. interpreted languages
- The steps of compiling: parsing, optimization, object code, linking, executable images
- How to write and compile a rudimentary C program
- Programming styles: imperative, structured, procedural, object-oriented, declarative, and functional
- The utility of stand-alone, distributed, client-server and web-enabled programming
- Statements, commands, comments and reserved words
- Routines, subroutines, functions and libraries
- Control flow structures
- Algorithms

Course Description

Courses that teach programming languages tend to focus on the particulars of each language (as they should), expending only minimal time discussing basic universal programming concepts. Students who are new to programming may find themselves engrossed in discussions of the details of matters such as control flow operators, executable images, third generation languages, and logical operators before they really understand the concepts. This situation can leave the student lost or frustrated or both.

This course introduces such universal programming concepts and is offered for the aspiring programmer so that they will never be left behind for want of having the foundational knowledge that is necessary to learn and advance.

Course Prerequisites

Rudimentary knowledge of computer systems. Students should know how to basically operate a computer, browse the world wide web, send and receive email messages, and write and print memos.

Course Overview

The Essence of Programming

- The language of computers
- Computer components
- Hardware and software
- Operating systems
- Programs

Types of Programming

- Computer languages
- Machine code
- Early programming
- The evolution of programming languages
- First Generation Languages
- Second Generation Languages
- Third Generation Languages
- Fourth Generation Languages
- Fifth Generation Languages
- Compilers
- The process of compiling
- Practical compiling
- Interpreted languages
- Pros and cons of compiled and interpreted languages
- Imperative programming
- Structured programming
- Procedural programming
- Object-oriented programming
- Declarative programming
- Functional programming
- Stand-alone programs
- Distributed programming
- Client-server programming
- Web-enabled programming
- Textual programming
- IDEs
- Visual programming

The Tasks of Programmers

- Analysis and requirements gathering
- Design and planning
- Coding
- Testing
- Deployment/Documentation
- Maintenance
- Your job

Statements and Storage

- The organization of programs
- Libraries
- Declaring and initializing variables and constants
- Data types: integers, floats,

Operators and Control Flow

- Assignment operators
- Arithmetic operators
- Relational operators
- Conditional operators
- Logical operators

Booleans, characters, arrays, structures, objects, pointers
- Proper data typing
- Allocation
- Static and dynamic data typing
- Strong and weak data typing
- Scope
- Inheritance

- Bitwise operators
- Special operators
- if, if-then, if-then-else statements
- case or switch-case statements
- for or foreach statements
- while statements
- do-while statements

The Language of Programming

-
- Algorithms
- Strings
- Regular expressions
- Input/Output
- Graphical user interfaces
- Events
- Frameworks
- Debugging
- Compile time vs. run time errors
- Database programming
- Technical communities
- Benchmarks
- Protocols

The Real World

- Working with the Command Prompt
- Customizing your environment
- Help
- Online documentation
- Batch files and scripts
- Installing and using a compiler
- Man pages
- Writing a C program
- Debugging C programs