

Comprehensive Programming in the C Language

Course Summary

Description

This course will teach the syntax and constructs of the ANSI C programming language. Programs in the course are designed to work in a(ny) Unix, Windows, OpenVMS, and other operating system platforms that support a C language compiler.

The overall course objective is to introduce basic and intermediate level concepts and techniques of the ANSI C programming language.

Topics

- C programming structure
- C data types
- Basic operators
- Beginning input and output in C
- Commands for C program development
- Loop constructs
- Arrays
- Decision statements
- Defining and calling functions
- Character string operations
- Structures
- Defining macros
- C programming standards

Audience

This course is designed for those wanting to learn basic and intermediate level concepts and techniques of the ANSI C programming language.

Prerequisites

It is assumed that the participant has used an interactive computer system, specifically, use of a text-file editor. Basics of the specific editor(s) used for the course will be covered during the program development module.

Duration

Four or five days

Comprehensive Programming in the C Language

Course Outline

- I. C programming structure**
 - A. main routine layout
- II. C data types**
 - A. statements used to allocate storage
 - B. radix specifiers
- III. Basic operators**
 - A. unary
 - B. binary
 - C. logical
 - D. conditional
- IV. Beginning input and output in C**
 - A. printf function
 - B. scanf function
- V. Commands for C program development**
 - A. compile
 - B. link
 - C. run
- VI. Loop constructs**
 - A. for statement
 - B. while statement
 - C. do while statement
- VII. Arrays**
 - A. layout
 - B. dimensioning
 - C. accessing
- VIII. Decision statements**
 - A. if blocks
 - B. switch block
- IX. Defining and calling functions**
 - A. prototyping
 - B. data declarations
 - C. returning values
- X. Character string operations**
 - A. null-terminated strings
 - B. C run-time string functions
- XI. Structures**
 - A. struct statement
 - B. accessing structure members
 - C. structures containing arrays
 - D. structures containing structures
- XII. Defining macros**
 - A. pre-processor directives
 - B. including directives at compile time
 - C. conditional compilation
- XIII. C programming standards**
 - A. C89 and C99 standards considerations