C Programming for Experienced Programmers

Course 6065 – 40 Hours

Overview

C is one of the most widely-used languages for systems software and workstation application programming, largely due to its power and flexibility. This course will provide you a highly effective, structured approach to learning ASCI Standard C revision C90 for experienced programmers who have had some exposure to similar languages such as Java or C#.
Programming skills will be enhanced as delegates will be able to use the powerful features of C to best effect building on the skill gathered from their previous experience in similar high level languages. This C Programming Course for Experienced Programmers is outstanding because of its emphasis on writing style, pitfalls to avoid and techniques to use that make the code clear, concise and maintainable. It is designed to accelerate through the foundations of C programming as appropriate to the skill level.
In addition to the lecture material there are graded practical sessions that cover each of the major areas of C. There are also optional exercises for further study after the course.

On Completion, Delegates will be able to

- Use all the major elements of the C language
- Write programs using the strengths of the C language, for example aggregate types, pointers and dynamic data structures
- Write and use the data structuring features of the language, which can result in better program design and maintenance
- Work with the C run-time library: a major source of programmer productivity and portability
- Dealing with errors, both by spotting and remedying common programming errors and using C's error handling facilities
- Write in a good C programming style without compromising efficiency and integrity

Who Should Attend

- Experienced programmers wishing to learn the C language.

Prerequisites

- Delegates must have professional programming skills and a good working knowledge of a C-like language together with the usual skills in a block-structured language, such as Basic/Visual Basic, Pascal/Modula2, Fortran, Algol or PL/1, and be familiar with a programming environment.

JOHN BRYCE
Leading in IT Education
a matrix company

matrix

Course Contents

An Overview of C
- History and evolution of C
- Key characteristics of C

Writing a Simple Program
- Program structure
- Data and code statements
- C software development life cycle

Data Types, Operators and Expressions
- Scalar types
- Variables and constants
- Storage considerations
- Initialising variables
- Standard arithmetic operators
- Increment, decrement, assignment and relational operators
- Automatic and programmer-controlled type conversion
- Introduction to evaluation points

Control Flow
- Boolean expressions
- While, Do and For loops
- Looping style considerations
- If, Else and Switch statements
- Other statements affecting flow of control
- Decision-making style considerations

Functions and Program Structure
- Inter-function communication
- Function prototypes, calls and definitions
- Scope and storage classes, stack
- Recursion

Structured Data Types
- Arrays, strings, structures and unions
- Nested data structures

Pointers
- The concept of indirection
- Pointers and address arithmetic
- Pointers and functions
- pointers and const

Pointers and Data Structures
- Pointers and arrays

- Pointers and structures
- Linked Lists
- Overview of Queue and Tree Structures

Preprocessor

- Tokens and macros
- Include files
- Conditional compilation, File Guarding
- Operators

Input and Output

- Using run-time routines
- Character and formatted I/O
- File I/O
- Block I/O

Further Data Types, Pointers and Error Handling

- Bit manipulation
- User-defined types
- Function Pointers, asserts and errno

Working with Larger Programs

- C and modular programming
- Linkage
- Cs standard library

The Way Ahead

- Further C information sources