

INTRODUCTION TO C LANGUAGE

Duration: 5 days

Course Summary: This course is for programmers who need a detailed and thorough introduction to the basics of C language. More than 150 complete sample programs with input and output as well as a case study will assist students in mastering all topics quickly. Students will create moderate level application programs using compiled statements, preprocessor directives, and calls to standard library functions. Three optional units are covered if there is time.

Prerequisites: Experience with any programming language.

Format: 50% lecture, 50% hands-on lab exercises.

Topics Covered In This Course:

Unit 1, Overview Of C Language

Overview of C Language

Documentation

First Program: Three Ways to Say Hello

Notes, First Program

Compiling in Unix, Linux, and Windows

Unit 2, Identifiers, Basic Data Types, Constants, Declarations, Printf

Identifiers, Keywords

Basic Data Types

Constants

Declarations and Actions

Printf Conversion Specifications

Ascii Character Code

Unit 3, Expressions, Operators, And Scanf

Expressions, Precedence and Associativity of Operators

Arithmetic Operators

Scanf Conversion Specifications

Assignment Operators

Comparison Operators

Sizeof Compile-Time Operator

Cast Operator

Type Conversion

Ternary Operator Aka Conditional Expression

Nested Ternary Operator

Unit 4, Compound Statements And Conditionals

Compound Statement, Aka Block

If

Switch

Unit 5, Loops, Break, And Continue

While

Do While

For

Break And Continue

Unit 6, Arrays, Getchar, And Putchar

Declaring an Array and Referencing an Element

Two- and Three-Dimensional Arrays

Two Views of a Two-Dimensional Array

Initializing Arrays

Getchar, EOF

Putchar

Unit 7, Introduction To Pointers

Declare, Reference, Dereference a Pointer

Null Pointers

Pointers and Arrays, Scaling

Unit 8, Strings And String Functions

Strings

Printf with %s

Scanf %s

Gets and Puts

Fgets and Fputs

Str... Functions Strlen, Strcpy, Strcat, Strcmp

Standard Library Functions to Process Char Arrays

Unit 9, User-Defined Functions

Defining Functions

Arguments, Parameters, and Return Values

Return Statement

Exit Function

Declarations and Prototypes of Functions

Prototypes, Void, and Argument Evaluation

Addresses and Pointers as Arguments

Pointer and Array Notation in Parameters

Unit 10, Storage Classes

The Dynamic Area

External Variables, Extern Declarations

The Data Area and Static Variables

Solving a Problem with Pointers

Unit 11, Structures, Unions, Typedef

Structure Definition and Declaration

Padding Bytes in RAM

Members in a Structure, Padding

Initializing and Assigning a Structure

Pointers to Structures

Function Calls with Structures

Nested Structures

Unions

Typedef

Optional Unit 12, The Preprocessor

The Preprocessor

Symbolic Constants, #Define

Conditional Compilation

Two Rules for Macros

Comparison of a Macro to a Function Call

Including Header Files

Assertions

Unit 13, File I/O

Preprocessor Terms Used In I/O

Fopen, Fclose

Character I/O, Getchar, Putchar, Getc, Putc, Fgetc, Fputc

Formatted I/O, Printf, Scanf, Fprintf, Fscanf, Sprintf, Sscanf

Binary I/O, Fread, Fwrite

Optional Unit 14, Pointers To Pointers

Double Indirection

Pointers to Strings

Array of Pointers to Arrays and Strings

Argc, Argv

Optional Unit 15, Memory Allocation And Deallocation

The System Function

Memory Allocation and Reallocation

Free Allocated Space

Suggestions for Further Study