DEV400

INTRODUCTION TO PROGRAMMING C

DEV 400 - Introduction to Programming C offers a fast-paced computer science course for students without programming experience. Students learn to write in the language used for operating systems, embedded processors, micro-controllers, assemblers, exploits and network drivers. This course focuses on solving problems using C, by teaching its fundamental principles, and techniques used in software engineering.

DEV400 assumes no prior programming experience.

TARGET AUDIENCE

Individuals with or without programming experience

OBJECTIVE

Provide an introduction to writing language for operating systems, embedded processors, micro-controllers, assemblers, exploits and network drivers

CYBRScore™

DAY 1

Day 1 begins by introducing how computers work and how C programs are processed by a computer. Students learn how to apply the software development model to solve programming problems. The format of C programs, data types, and variables is discussed, and students also learn how to program arithmetic expressions, assign values to variables, and read data into a program and display the results. Students learn about top-design to develop algorithms and program structure. Self-check exercises and labs challenge students to use the newly-learned information in context.

Topics List

- Operating system, machine language, assembly language and high-level languages
- Format and elements of a C program
- Various data types
- Arithmetic expressions and problem solving
- Format strings
- Functions
- Modular programming
- Common programming errors

DAY 2

Day 2 discusses control structures and how to manipulate program execution and control flow. Students learn to make comparisons and code the logic behind conditionals. The labs challenge students to use a variety of control structures like sequence, selection, and repetition.

Topics List

- Control structures
- Compare numbers, compare characters and strings
- "If" statements
- Decision algorithms
- Switch statement and various alternatives
- Case Study: Utility Bill; Problem: Write a program to compute a customer's utility bill based on numerous factors

DAY 3

Students learn why repetition is an important concept in programming. The Repetition and Loops lectures and labs teach students how to use repetition in their programs to gain efficiencies. Counting, sentinel-controlled and flag-controlled loops and the usefulness of nested loops are discussed. Traditional for, while, and do-while statements for creating loops are covered in-depth.

Topics List

- Loop design and infinite loops
- Counting Loops, sentinel-controlled loops, flag-controlled loops
- For, while, do-while statements
- Nested loops
- Outer and inner loop control variables
- Case Study: Bisection Method for Finding Roots; Problem: Develop a function to approximate the root of a mathematical function containing an odd number of roots

DAY 4

The Pointers lectures and labs expose students to indirect addressing. Reading from and writing to files using file pointers and comparing call-by-value and call-by-reference methods are discussed. Finally the student will learn how to pass information to and get information back from functions.

Topics List

- Pointers
- Indirect addressing
- File I/O using pointers
- Call methods for functions
- Modularize program with functions
- Working with pointers and functions
- Case Study: Solar-heated House; Problem: Design a program to estimate size of collecting area needed to support a solar-heated house

DAY 5

The Array Pointers lectures and labs teach students how to use arrays and how C implements arrays as pointers. Students learn how to access values in arrays, and how to process data in arrays using loops. Searching arrays is introduced and multi-dimensional arrays are described as a solution for storing tables of data.

Capstone Exercise

- Arrays
- Relationship between arrays and pointers
- Process array data with loops
- Function and array interaction
- Searching and sorting arrays
- Case Study: Summary of Business Revenue; Problem: Regional medical center needs software to track its revenue by unit and quarter

DEV400

INTRODUCTION TO PROGRAMMING C

CYBRScore™