About Codefrux

While the current trends around the world are based on the internet, mobile and its applications, we try to make the most out of it. As for us, we are a well established IT professionals based in Bangalore, constantly coping up with the extensive advancement and adapting to new Technology.

C is a powerful general-purpose programming language. It is fast, portable and available in all platforms. If you are new to programming, C is a good choice to start your programming journey. This is a comprehensive guide on how to get started in C programming language, why you should learn it and how you can learn it..

What you will Learn In This Course

- Master C programming concepts from the ground up
- Write code and learn step-by-step
- Understand the special features of C: pointers, header files, null-terminated strings, buffers, IO
- Explore the topics in even more depth
- Test your understanding with end-of-section quizzes
- Arrays and ArrayLists

Who should take the course

- Beginners – if you've never coded before, you can learn C step by step
- Programmers switching to C from some other language such as Java, Ruby or Python
- Cross-platform developers – there are C compilers for all major operating systems
- Anyone who needs to program C++ or Objective-C. The C language is the place to start

1. **C Programming Introduction**
    1. Facts about C
    2. Why Use C?
    3. The C Language and its Advantages
    4. C Programs
    5. Quiz
    6. Summary
    7. Hands on

2. **Environment Setup**
    1. Try it Option Online
    2. Local Environment Setup
    3. Text Editor
    4. The C Compiler
    5. Compile-Time Directives
    6. Use of typedef
    7. C Preprocessor Syntax
    8. Installation on Windows
    9. Quiz
    10. Summary
    11. Hands on

3. **Program Structure**
    1. Hello World Example
    2. Compile and Execute C Program
    3. Quiz
    4. Summary
    5. Hands on

4. **Basic Syntax**
    1. Tokens in C
    2. Semicolons
    3. Comments
    4. Identifiers
    5. Keywords
    6. Whitespace in C
    7. Quiz
    8. Summary
    9. Hands on

5. Data Types

1. Integer Types
2. Floating-Point Types
3. The void Type
4. Quiz
5. Summary
6. Hands on

6. Variables

1. Variable Definition in C
2. Variable Declaration in C
3. Lvalues and Rvalues in C
4. Quiz
5. Summary
6. Hands on

7. Constants And Literals

1. Integer Literals
2. Floating-point Literals
3. Character Constants
4. String Literals
5. Keywords
6. Defining Constants
7. The #define Preprocessor
8. The const Keyword
9. Quiz
10. Summary
11. Hands on

8. Storage Classes

1. The auto Storage Class
2. The register Storage Class
3. The static Storage Class
4. The extern Storage Class
5. Quiz
6. Summary
7. Hands on

9. Operators

1. Arithmetic Operators
2. Relational Operators
3. Logical Operators
4. Bitwise Operators
5. Assignment Operators

6. Misc Operators ↦ sizeof & ternary
7. Operators Precedence in C
8. Quiz
9. Summary
10. Hands on

10. Decision Making

1. if Statement
2. if…else Statement
3. if else if else Statement
4. Nested if Statements
5. switch Statement
6. Nested switch Statements
7. The ? : Operator
8. Quiz
9. Summary
10. Hands on

11. Loops

1. while Loop
2. for Loop
3. do…while Loop
4. Nested Loops
5. Loop Control Statements
6. break Statement
7. continue Statement
8. goto Statement
9. The Infinite Loop
10. Quiz
11. Summary
12. Hands on

12. Functions

1. Defining a Function
2. Function Declarations
3. Calling a Function
4. Function Arguments
5. Call by Value
6. Call by Reference
7. Quiz
8. Summary
9. Hands on

13. Scope Rules

1. Local Variables
2. Global Variables
3. Formal Parameters
4. Initializing Local and Global Variables
5. Quiz
6. Summary
7. Hands on

14. Arrays

1. Declaring Arrays
2. Initializing Arrays
3. Accessing Array Elements
4. Arrays in Detail
    a. Multidimensional Arrays
    b. Two-dimensional Arrays
    c. Initializing Two-Dimensional Arrays
    d. Accessing Two-Dimensional Array Elements
    e. Passing Arrays to Functions
    f. Return Array from a Function
    g. Pointer to an Array
5. Quiz
6. Summary
7. Hands on

15. Pointers

1. What are Pointers?
2. How to Use Pointers?
3. NULL Pointers
4. Pointers in Detail
    a. Pointer Arithmetic
    b. Incrementing a Pointer
    c. Decrementing a Pointer
    d. Pointer Comparisons
    e. Array of Pointers
    f. Pointer to Pointer
    g. Passing Pointers to Functions
    h. Return Pointer from Functions
5. Quiz
6. Summary
7. Hands on

16. Strings and Character Manipulation

1. Strings as Character Arrays
2. String Library Functions
3. Reading and Writing Strings
4. Quiz
5. Summary
6. Hands on

17. Structures

1. Defining a Structure
2. Accessing Structure Members
3. Structures as Function Arguments
4. Pointers to Structures
5. Bit Fields
6. Quiz
7. Summary
8. Hands on

18. Unions

1. Defining a Union
2. Accessing Union Members
3. Quiz
4. Summary
5. Hands on

19. Bit Fields

1. Bit Field Declaration
2. Quiz
3. Summary
4. Hands on

20. Typedef

1. typedef vs #define
2. Quiz
3. Summary
4. Hands on

21. Input And Output

1. The Standard Files
2. The getchar() and putchar() Functions
3. The gets() and puts() Functions
4. The scanf() and printf() Functions
5. Quiz
6. Summary

7. Hands on

22. File I/O

1. Opening Files
2. Closing a File
3. Writing a File
4. Reading a File
5. Binary I/O Functions
6. Quiz
7. Summary
8. Hands on

23. Preprocessors

1. Preprocessors Examples
2. Predefined Macros
3. Preprocessor Operators
4. The Macro Continuation (\) Operator
5. The Stringize (#) Operator
6. The Token Pasting (##) Operator
7. The Defined() Operator
8. Parameterized Macros
9. Quiz
10. Summary
11. Hands on

24. Header Files

1. Include Syntax
2. Include Operation
3. Once-Only Headers
4. Computed Includes
5. Quiz
6. Summary
7. Hands on

25. Type Casting

1. Integer Promotion
2. Usual Arithmetic Conversion
3. Quiz
4. Summary
5. Hands on

26. Error Handling

1. errno, perror(), and strerror()
2. Divide by Zero Errors
3. Program Exit Status

4. Quiz
5. Summary
6. Hands on

27. Recursion

1. Number Factorial
2. Fibonacci Series
3. Quiz
4. Summary
5. Hands on

28. Variable Arguments

6. Quiz
7. Summary
8. Hands on

29. Memory Management

1. Quiz
2. Summary
3. Hands on

30. Command Line Arguments

1. Quiz
2. Summary
3. Hands on

Project Work

After course completion, students will be assigned to work on live project to polish the technology skills you have acquired with us.