**Total Duration: 30 Hours (12-14 Classes)**

C Introduction

* [Keywords & Identifier](https://www.programiz.com/c-programming/c-keywords-identifier)
* [Variables & Constants](https://www.programiz.com/c-programming/c-variables-constants)
* [C Data Types](https://www.programiz.com/c-programming/c-data-types)
* [C Input/Output](https://www.programiz.com/c-programming/c-input-output)
* [C Operators](https://www.programiz.com/c-programming/c-operators)
* [C Introduction Examples](https://www.programiz.com/c-programming/c-introduction-examples)

C Flow Control

* [C if...else](https://www.programiz.com/c-programming/c-if-else-statement)
* [C for Loop](https://www.programiz.com/c-programming/c-for-loop)
* [C while Loop](https://www.programiz.com/c-programming/c-do-while-loops)
* [C break and continue](https://www.programiz.com/c-programming/c-break-continue-statement)
* [C switch...case](https://www.programiz.com/c-programming/c-switch-case-statement)
* [C Programming goto](https://www.programiz.com/c-programming/c-goto-statement)
* [Control Flow Examples](https://www.programiz.com/c-programming/c-decision-making-loops-examples)

C Functions

* [C Programming Functions](https://www.programiz.com/c-programming/c-functions)
* [C User-defined Functions](https://www.programiz.com/c-programming/c-user-defined-functions)
* [C Function Types](https://www.programiz.com/c-programming/types-user-defined-functions)
* [C Recursion](https://www.programiz.com/c-programming/c-recursion)
* [C Storage Class](https://www.programiz.com/c-programming/c-storage-class)
* [C Function Examples](https://www.programiz.com/c-programming/c-functions-examples)

C Programming Arrays

* [C Programming Arrays](https://www.programiz.com/c-programming/c-arrays)
* [C Multi-dimensional Arrays](https://www.programiz.com/c-programming/c-multi-dimensional-arrays)
* [C Arrays & Function](https://www.programiz.com/c-programming/c-arrays-functions)

C Programming Pointers

* [C Programming Pointers](https://www.programiz.com/c-programming/c-pointers)
* [C Pointers & Arrays](https://www.programiz.com/c-programming/c-pointers-arrays)
* [C Pointers And Functions](https://www.programiz.com/c-programming/c-pointer-functions)
* [C Memory Allocation](https://www.programiz.com/c-programming/c-dynamic-memory-allocation)
* [Array & Pointer Examples](https://www.programiz.com/c-programming/c-pointer-examples)

C Programming Strings

* [C Programming String](https://www.programiz.com/c-programming/c-strings)
* [C String Functions](https://www.programiz.com/c-programming/string-handling-functions)
* [C String Examples](https://www.programiz.com/c-programming/c-string-examples)

Structure And Union

* [C Structure](https://www.programiz.com/c-programming/c-structures)
* [C Struct & Pointers](https://www.programiz.com/c-programming/c-structures-pointers)
* [C Struct & Function](https://www.programiz.com/c-programming/c-structure-function)
* [C Unions](https://www.programiz.com/c-programming/c-unions)
* [C struct Examples](https://www.programiz.com/c-programming/c-structure-examples)

C Programming Files

* [C Files Input/Output](https://www.programiz.com/c-programming/c-file-input-output)
* [C Files Examples](https://www.programiz.com/c-programming/c-file-examples)

Additional Topics

* [C Enumeration](https://www.programiz.com/c-programming/c-enumeration)
* [C Preprocessors](https://www.programiz.com/c-programming/c-preprocessor-macros)
* [C Standard Library](https://www.programiz.com/c-programming/library-function)
* [C Programming Examples](https://www.programiz.com/c-programming/examples)

## About C Programming

* **Procedural Language** - Instructions in a C program are executed step by step.
* **Portable** - You can move C programs from one platform to another, and run it without any or minimal changes.
* **Speed** - C programming is faster than most programming languages like Java, Python, etc.
* **General Purpose** - C programming can be used to develop operating systems, embedded systems, databases, and so on.

## Why Learn C Programming?

* C helps you to understand the internal architecture of a computer, how computer stores and retrieves information.
* After learning C, it will be much easier to learn other programming languages like Java, Python, etc.
* Opportunity to work on open source projects. Some of the largest open-source projects such as Linux kernel, Python interpreter, SQLite database, etc. are written in C programming.