

---

## Basic C

---

If you are a student wanting to get into software development, or an professional wanting to add C programming skills to your repertoire, this introductory course is for you.

The C programming language is a popular and widely used programming languages. It is a general-purpose programming language. There would be very few computers in the world which are not setup for C.

This course introduces you to the basics of programming in C. You will learn how programming languages work with data, what program flow is, and how to use functions, methods and routines. You will also get training in on how to create simple C programs and run them.

---

### Why Need It

---

This course will be useful to anybody who wants to go into IT, software and computer industry.

An introduction to the C programming language is the FIRST step to quickly grasp the intricacies of object oriented programming (with programs like C++) and program in other languages which have been influenced by C, like Java & JavaScript, to name a few.

---

### Features of the Course

---

- Conducive Environment for new programming students
- Experienced Faculty
- Lots of Practise to get to the nitty gritty of problems faced with C programming

---

### Course Goals

---

- Learn Object Oriented Programming & how to program in C
- Learn about program flow and Object Oriented Programming
- Learn about functions, methods and routines & How to use arguments and return values
- Run a simple C program.

---

### Who Can Take It

---

Anybody with a minimal knowledge of computers and education upto Class Xth.

---

# Course curriculum

---

---

## 1. Overview of C

---

- Operating System Independence
- Design Goals and Capabilities
- Flavors of C

---

## 3. Macros

---

- Macros
- Functions versus Inlining
- Purpose of Macros
- Use of Macros
- Making Code More Readable
- Auto Adjustment of Compile Time Values
- Conditional Compilation
- Making Code Portable
- Simplifying Complex Access Calculations
- Advanced Micro Design Tips
- Using Macros to Help Write Portable Programs
- When to Use a Macro instead of a Function
- Using Macros for Debugging

---

## 5. Operators and Expressions

---

- Arithmetic, Logical, and Bit Operators

---

## 2. Fundamental Data Types, Storage Classes, and Scope

---

- Fundamental Data Types and Qualifiers
- Constants and Strings
- Storage Classes
- Scope and Block Structure
- Scope and Data Hiding
- Data Initialization

---

## 4. Basic Formatted I/O

---

- Standard I/O Library
- Character Set Encoding
- Standard Input and Output
- Character I/O Functions
- Formatted I/O Functions
- String Constants

---

## 6. Functions (Subroutines)

---

- Purpose of Functions
- Functions versus Inlining
- Automatic Variables
- The Argument Stack
- Passing By Value
- Passing By Reference
- Declaring External Functions
- Function Prototyping

- Precedence and Associativity
- Assignment and Casting
- The Conditional Operator

- ANSI Prototyping
- The `_NO_PROTO` Compiler Symbol
- Varargs Functions
- Passing a Function as an Argument
- Designing Functions for Reusability
- Calling a Function from Another Language
- Returning a Dynamically Allocated Value Using Double Indirection
- Casting the Return Value of a Function
- Recursion and Reentrancy

---

## 7. Advanced Structures and Unions

---

- Nested Structures
- Arrays of Structures
- Bit Fields
- Unions
- Linked Lists

---

## 8. Strings and Character Manipulation

---

- Strings as Character Arrays
- String Library Functions
- Reading and Writing Strings

---

## 9. Structured Programming

---

- Structuring Code for Quality, Reliability, Maintainability
- Designing for Modularity and Reusability

---

## 10. Compiler Directives and the C Preprocessor

---

- Compile-Time Directives
- Use of `typedef`
- C Preprocessor Syntax

---

## 11. Pointers and Dynamic Allocation

---

- Advantages of Pointers
- User of Pointers
- Pointer and Address Arithmetic
- Dynamic Storage Allocation
- `sizeof`
- Double Indirection

---

## 12. Arrays

---

- Purpose of Arrays
- Declaring an Array
- Initializing an Array
- Addressing Elements
- Stepping Through an Array
- Variable Size Arrays
- Arrays of Pointers
- Arrays of Strings
- Passing an Array to a Function
- Dynamic Memory Allocation
- Multidimensional Arrays

---

## 13. Program Debugging

---

- Problem Analysis

---

## 14. Flow Control Constructs

---

- Instrumenting with printf
- Instrumenting with ctrace
- The Purpose of Debuggers
- How Not to Use Debuggers
- Symbolic Debuggers

- Conditional Constructs: if, switch
- Looping Constructs: while, do, for
- Programming Style

---

## 15. Structures

---

- Purpose of Structures
- Defining and Declaring Structures
- Accessing Members
- Pointers to Structures
- Dynamic Memory Allocation
- Passing a Structure to a Function
- As a Pointer
- Passing the Actual Structure

---

## 16. C Runtime Library Standard Functions

---

- Character I/O
- Unformatted File I/O
- Formatted File I/O
- Math Functions
- Miscellaneous Functions

---

## 17. Accessing Command Line Arguments and Environment Symbols

---

- argc and argv
- Parsing Command Line Options
- Accessing the Environment Array

---

## 18. Advanced Programming Consideration

---

- Writing Portable Code
- Use of Macros
- ANSI C Limits
- Feature Test Macros
- Client/Server Design
- Performance Considerations

[Enroll for this batch](#)

We always keep ourselves updated about current market trends & needs, but if you think we have not covered enough training topics, then we welcome you to suggest us & we will be more than happy to include them in syllabus.

**Customized Corporate Private Trainings:** We deliver all our courses as [Corporate Training](#) as well - if you are a group interested in the course, this option may be more advantageous for you.

---

## Other courses

---

[Web Design](#)

[HTML5](#)

[JavaScript](#)

[PHP](#)

[Front end development](#)

[Wordpress](#)

[Drupal](#)

[Joomla](#)

[WPE](#)

[NodeJS](#)

[AngularJS](#)

[BackBone JS](#)

[SEO](#)

[SMO](#)

[See All »](#)

---

## Follow us on Social Media

---

- [Find Technnovation on Facebook](#)
- [Follow Technnovation on Twitter](#)
- [Follow Technnovation on LinkedIn](#)
- [Google plus Technnovation](#)

---

## Contact Us

---

4th Floor, Nandlal Hsg Society,  
418 Narayan peth, Above Bedekar misal,  
Munjobacha Bol, Shagun Chowk, Pune, Maharashtra - 411030  
India, Contact: **898 300 2500 / 3700**  
<http://www.tlabsonline.com>  
[Live Chat support](#)  
[Earn back](#)