

C Programming Language (C99)

Why C

The C Language is currently one of the most widely used programming languages in the world. Designed as a tool for creating operating systems (the first Unix systems were constructed with its help), it quickly proved that it is suitable wherever you need high performance, speed, compactness and portability. Even though shortly after its release it was followed by a worthy descendant, the C++ language, it did not lose its importance, and it remains an essential tool for developers and designers in many applications.

Whenever a code strongly associated with the operation of equipment is created, the C language proves its flexibility and adaptability. Network card drivers, graphics card software, operating systems and micro-controllers – these things can be found all around you, on your desk and in your car, in the kitchen and in the garage, simply everywhere intelligent electronics functions – you are sure to find the work of programmers who write in “C”. The heart of Linux is nearly 15 million lines of code in “C”. There is no better evidence for the language’s longevity.

Even in places where modern software with much more powerful abilities works, the C language was, is and will be present, because it is the language in which runtimes (runtime environments) are written, responsible for performance, economical memory usage and reliability. The “C” language niche extends from single-chip microcomputers controlling your coffee machine, to your laptop onto which you have just installed the latest graphics card drivers, to supercomputers that forecast the weather for your desired holiday.

Syllabus

- Evolution of C – from past to eternity
 - milestones: ANSI C, C89, C95, C99, C11
 - obsolete (but still valid) language elements
 - how have function declarations changed over time?
 - trigraphs and digraphs
 - new C11 keywords:
 - `_Noreturn`, `_Alignof` and `_Alignas`, `_Bool`
 - `_Exit`, `_Complex`, `_Pragma`, `__func__`, `_Generic`
 - Strengths and Weaknesses of C
- Steps involved in Developing a C application
- Use Integrated Development Environments
- General Layout of a C Program
 - Directives
 - Functions
 - Statements
 - Printing Strings
- Basic Types
 - Character Types
 - Integer Types
 - Integer Overflow
 - Floating Types

Impetus IT Services Pvt.Ltd.

B-16, First floor, Sant Tukaram Vyapar Sankul, Sector - 24, Nigdi, Pune, Maharashtra. India. Pin – 411044.

Mobile 9970600774, 9730012775 | Board 91-20-27640406 | Fax 91-20-27641703

Email : hrishikesh@impetusitservices.com | Website <http://impetusits.in>

- Data type modifiers
- Keywords
- Variables
- Constants
 - Types of C Constants
- Input Output Operations in C
 - Console Input and Output
 - Concept of streams
 - String input and output function
 - Formatted input and output functions
 - Format modifiers
- Operators in C
- Bitwise Operators in C
- Selection constructs
 - If Construct
 - Nested If Construct
 - Switch Case Construct
- Iteration Constructs
 - For loop
 - While loop
 - Do while loop
- Jump Statements
 - Return statement
 - Break statement
 - Continue statement
 - Exit () function
 - Go to & labels
- Functions
 - What is a function?
 - Why use function
 - Function Prototype
 - Return statement
 - Actual and formal arguments
 - Local variables
 - Global variables
 - Invoking Function
 - Call by Value
 - Call by Reference
 - Recursive Function
 - General Purpose function
 - Inline code
 - Function returning a pointer
 - Function as argument to function

Impetus IT Services Pvt.Ltd.

B-16, First floor, Sant Tukaram Vyapar Sankul, Sector - 24, Nigdi, Pune, Maharashtra. India. Pin – 411044.

Mobile 9970600774, 9730012775 | Board 91-20-27640406 | Fax 91-20-27641703

Email : hrishikesh@impetusitservices.com | Website <http://impetusits.in>

- Preprocessor Directives
 - Features of C Preprocessor
 - Macro expansion
 - File inclusion
 - # Include
 - # define
 - Conditional Compilation Directives
 - # if, # else, #elif, #endif, # ifdef, #ifndef, #undef
 - #pragma (intro)
 - Macros versus functions
- Storage Classes
 - Lifetime and visibility of variables
 - Automatic storage class
 - External storage class
 - Static storage class
 - Register storage class
- Arrays
 - What are arrays?
 - Defining an array
 - Subscripts and dimensions
 - Array Initialization
 - Single dimensional and two-dimensional arrays
 - String / Character arrays
- Common string functions
- Multidimensional Arrays
- Arrays as arguments to function
- Pointers
 - What are pointers?
 - Use of Pointers
 - Pointer variables
 - Pointer operators
 - Pointer Arithmetic
 - Pointer Comparison
 - Pointers as argument to function
 - Array using pointers (single dimensional and two dimensional)
 - Array of pointers
 - Pointer to an array
 - Multiple indirection
- Pointer to a function

Impetus IT Services Pvt.Ltd.

B-16, First floor, Sant Tukaram Vyapar Sankul, Sector - 24, Nigdi, Pune, Maharashtra. India. Pin – 411044.

Mobile 9970600774, 9730012775 | Board 91-20-27640406 | Fax 91-20-27641703

Email : hrishikesh@impetusitservices.com | Website <http://impetusits.in>

- Memory and strings (<string.h> et al.)
 - manipulating memory blocks
 - aligned_alloc(), calloc(), malloc(), realloc(), free()
 - string manipulation:
 - strchr(), strrchr(), strstr(), strtok(),
 - qsort(), bsearch()
 - bcopy(), memcpy(), memccpy(), memmove(), bzero(), memset(), memcmp()
- Structures and Unions
 - Defining a structure
 - Declaring structure variables
 - Accessing structure elements
 - Initializing structures
 - Assignment statements used with structures
 - Passing structures as arguments
 - Array of structures
 - Initialization of structure arrays
 - Pointers to structures
 - Structure pointers as arguments
 - Defining unions and accessing the elements
 - Unions of structures
- Typedef
- Enumeration
- File Input and Output
 - Streams and Files
 - Text and binary streams
 - Basic file functions
- Handling variable number of parameters (<stdarg.h>)
 - calling conventions,
 - passing parameters,
 - stack usage, stack frame,
 - returning a value,
 - va_start(), va_arg() , va_end(), va_copy()
- Sorting Techniques
 - Bubble Sort
 - Insertion sort
 - Selection sort
- Searching
 - Linear Searching
 - Binary Search

Impetus IT Services Pvt.Ltd.

B-16, First floor, Sant Tukaram Vyapar Sankul, Sector - 24, Nigdi, Pune, Maharashtra. India. Pin – 411044.

Mobile 9970600774, 9730012775 | Board 91-20-27640406 | Fax 91-20-27641703

Email : hrishikesh@impetusitservices.com | Website <http://impetusits.in>