C – Language History

The C programming language is a structure oriented programming language, developed at Bell Laboratories in 1972 by Dennis Ritchie

C programming language features were derived from an earlier language called “B” (Basic Combined Programming Language – BCPL)

C language was invented for implementing UNIX operating system

In 1978, Dennis Ritchie and Brian Kernighan published the first edition “The C Programming Language” and commonly known as K&R C

In 1983, the American National Standards Institute (ANSI) established a committee to provide a modern, comprehensive definition of C. The resulting definition, the ANSI standard, or “ANSI C”, was completed late 1988.

C programming language standards:

C89/C90 standard – First standardized specification for C language was developed by the American National Standards Institute in 1989. C89 and C90 standards refer to the same programming language.

C99 standard – Next revision was published in 1999 that introduced new features like advanced data types and other changes.

C11 and Embedded C language:

C11 standard adds new features to C programming language and library like type generic macros, anonymous structures, improved Unicode support, atomic operations, multi-threading and bounds-checked functions. It also makes some portions of the existing C99 library optional and improves compatibility with C++.

Embedded C includes features not available in C like fixed-point arithmetic, named address spaces, and basic I/O hardware addressing.

Operating systems, C compiler and all UNIX application programs are written in C language

It is also called as procedure oriented programming language. The C language is reliable, simple and easy to use. C has been coded in assembly language.

Features of C programming language:

Reliability

Portability

Flexibility

Interactivity

Modularity

Efficiency and Effectiveness

Uses of C programming language:

The C programming language is used for developing system applications that forms a major portion of operating systems such as Windows, UNIX and Linux. Below are some examples of C being used.

Database systems

Graphics packages

Word processors

Spreadsheets

Operating system development

Compilers and Assemblers

Network drivers

Interpreters

Which level is C language belonging to?

S.no

High Level

Middle Level

Low Level

1 High level languages provide almost everything that the programmer might need to do as already built into the language

Middle level languages don’t provide all the built-in functions found in high level languages, but provides all building blocks that we need to produce the result we want

Low level languages provides nothing other than access to the machines basic instruction set

2 Examples:

Java, Python C, C++ Assembler

The C language is a structured language

S.no

Structure oriented

Object oriented

Non structure

1 In this type of language, large programs are divided into small programs called functions In this type of language, programs are divided into objects There is no specific structure for programming this language

2 Prime focus is on functions and procedures that operate on the data Prime focus is in the data that is being operated and not on the functions or procedures N/A

3 Data moves freely around the systems from one function to another Data is hidden and cannot be accessed by external functions N/A

4 Program structure follows “Top Down Approach” Program structure follows “Bottom UP Approach” N/A

5

Examples:

C, Pascal, ALGOL and Modula-2

C++, JAVA and C# (C sharp)

BASIC, COBOL, FORTRAN

Key points to remember in C language:

The C language is structured, middle level programming language developed by Dennis Ritchie

Operating system programs such as Windows, Unix, Linux are written in C language

C89/C90 and C99 are two standardized editions of C language

C has been written in assembly language

C language tutorial reference E-books & research papers:

[ANSI 89] American National Standards Institute., American National Standard for Information Programming Language C, X3 159-1989

[Kernighan 78] B. W. Kernighan and D. M. Ritchie, The C Programming Language, Prentice-Hall: Englewood Cliffs, NJ, 1978. Second edition, 1988.

[Thinking 90] C\* Programming Guide, Thinking Machines Corp.: Cambridge Mass., 1990.

C – Basic Program

We are going to learn a simple “Hello World” C program in this section. Also, all the below topics are explained in this section which are the basics of a C program.

C basic program with output and explanation

Steps to write C programs and get the output

Creation, Compilation and Execution of a C program

How to install C compiler and IDE

Basic structure of a C program

Basic commands in C programming to write basic C Program:

Below are few commands and syntax used in C programming to write a simple C program. Let’s see all the sections of a simple C program line by line.

S.no Command Explanation

1 #include <stdio.h> This is a preprocessor command that includes standard input output header file(stdio.h) from the C library before compiling a C program

2 int main() This is the main function from where execution of any C program begins.

3 { This indicates the beginning of the main function.

4 /\*\_some\_comments\_\*/ whatever is given inside the command “/\* \*/” in any C program, won’t be considered for compilation and execution.

5 printf(“Hello\_World! “); printf command prints the output onto the screen.

6 getch(); This command waits for any character input from keyboard.

7 return 0;

This command terminates C program (main function) and returns 0.

8 }

This indicates the end of the main function.

1. C Basic Program:

#include <stdio.h>

int main()

{

/\* Our first simple C basic program \*/

printf(“Hello World! “);

getch();

return 0;

} .

Output:

Hello World! .

2.Steps to write C programs and get the output:

Below are the steps to be followed for any C program to create and get the output. This is common to all C program and there is no exception whether its a very small C program or very large C program.

C Basic program

3. Creation, Compilation and Execution of a C program:

Prerequisite:

If you want to create, compile and execute C programs by your own, you have to install C compiler in your machine. Then, you can start to execute your own C programs in your machine.

You can refer below link for how to install C compiler and compile and execute C programs in your machine.

Once C compiler is installed in your machine, you can create, compile and execute C programs as shown in below link.

C – Environment Setup Using IDE tool

C – Environment Setup Using GCC compiler

4. Basic structure of C program:

Structure of C program is defined by set of rules called protocol, to be followed by programmer while writing C program. All C programs are having sections/parts which are mentioned below.

Documentation section

Link Section

Definition Section

Global declaration section

Function prototype declaration section

Main function

User defined function definition section

Example C program to compare all the sections:

You can compare all the sections of a C program with the below C program.

/\* C basic structure program Documentation section

Author: fresh2refresh.com

Date : 01/01/2012

\*/#include <stdio.h> /\* Link section \*/

int total = 0; /\* Global declaration and definition section \*/

int sum (int, int); /\* Function declaration section \*/

int main () /\* Main function \*/

{

printf (“This is a C basic program \n”);

total = sum (1, 1);

printf (“Sum of two numbers : %d \n”, total);

return 0;

}

int sum (int a, int b) /\* User defined function \*/

{ /\* definition section \*/

return a + b;

} .

Output:

This is a C basic program

Sum of two numbers : 2 .

Description for each section of a C program:

Let us see about each section of a C basic program in detail below.

Please note that a C program mayn’t have all below mentioned sections except main function and link sections.

Also, a C program structure mayn’t be in below mentioned order.

S.No Sections Description

1 Documentation section We can give comments about the program, creation or modified date, author name etc in this section. The characters or words or anything which are given between “/\*” and “\*/”, won’t be considered by C compiler for compilation process.These will be ignored by C compiler during compilation.

Example : /\* comment line1 comment line2 comment 3 \*/

2 Link Section Header files that are required to execute a C program are included in this section

3 Definition Section In this section, variables are defined and values are set to these variables.

4 Global declaration section Global variables are defined in this section. When a variable is to be used throughout the program, can be defined in this section.

5 Function prototype declaration section Function prototype gives many information about a function like return type, parameter names used inside the function.

6 Main function Every C program is started from main function and this function contains two major sections called declaration section and executable section.

7 User defined function section User can define their own functions in this section which perform particular task as per the user requirement.

C programming tutorial reference E-books & research papers: