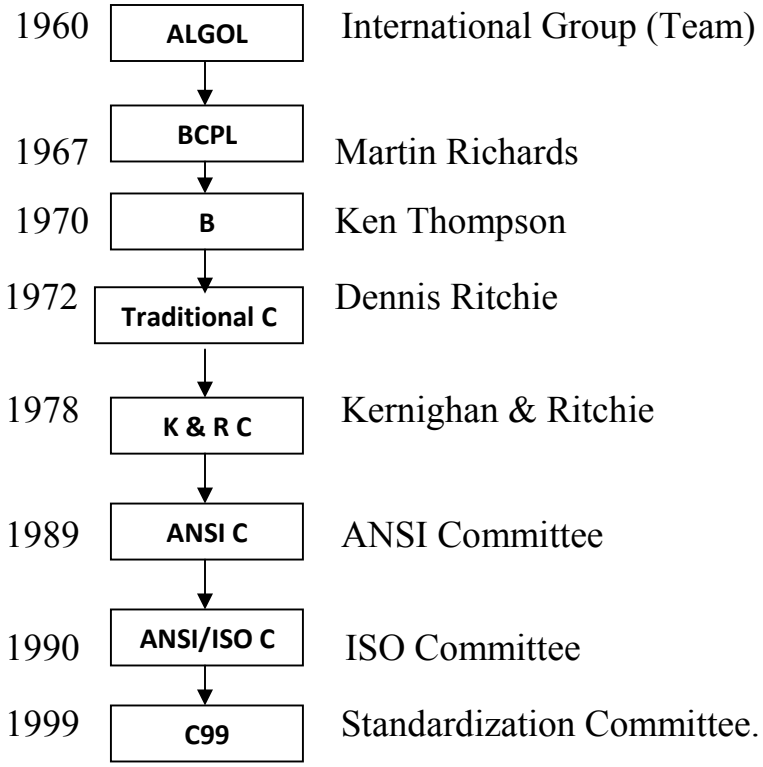
Question: Briefly discuss the history of C.

Answer: "C" seems a strange name for a programming language. But this strange sounding language is one of the most popular computer languages today because it is a structured, high level, machine independent language.



The root of all modern languages is ALGOL, introduced in the early 1960s. It was the first computer language to use a block structure. Although it never became popular in USA, it was widely used in Europe.

In 1967s, Martin Richards developed a language called BCPL (Basic Combined Programming Language) primarily for writing system software.

In 1970s, Ken Thompson created a language using many features of BCPL and called it simply B.

In 1972s, C was created from ALGOL, BCPL, and B by Dennis Ritchie.

In 1978s, Brain Kernighan & Dennis Ritchie was developed the C language which is called K & R C.

In 1989s, American National Standard Institute (ANSI) appointed a technical committee to define a standard for C. It is also known as: ANSI C.

In 1990s, The International Standards Organization (ISO) was approved the C, it is also known as: ANSI/ISO C.

In 1999s, The Standardization Committee was approved the C language which is called the C99.

Question: What is C language? Write some importance of C languages.

Answer: A C program is basically a collection of functions that are supported by the C library. We can continuously add our own functions to C library with the availability of a large number of functions, the programming task becomes simple.

There are some importances of a C programming language which is given below:

I. Reduce the number of lines in a code.
II. Making our code more readable.
III. Programs written in C are very efficient.
IV. Programs written in C are very fast.
V. C is highly portable.
VI. C is its ability to extend itself.

Question: Write a simple C program & explain it briefly.

Answer: Consider a very simple program given below:

```
Main ()
   {
      /*----printing begins----*?
     printf ("I See, I remember");
    /*----printing ends----*/
  };
```

When we execute this program, we will see the output is:

I See, I remember

The main () is a special function used by the C system to tell the computer where the program starts. Every program must have exactly one ***main*** function. If we use more than one ***main*** function, the compiler can't understand which one marks the beginning of the program. The ***main*** function is start with the opening brace & ends with the ending brace. The lines beginning with /* and ending with */ are known as ***comment lines***. Comment lines are not executable statements and therefore anything between /* and */ is ignored by the compiler. The ***printf*** function is used to write the output message. Every statement in C should end with a semicolon (**;**) mark.

Question: How many supports the main () function in C program?

Answer: The main is a part of every C program. C permits different forms of main statement. Following forms are allowed:

- ✓ Main()
- ✓ int main()
- ✓ void main()
- ✓ main(void)
- ✓ void main(void)
- ✓ int main(void)

The empty pair of parentheses indicates that the function has no arguments. The keyword **void** means that the function does not return any information to the operating system and **int** means that the function returns an integer value to the operating system.

Question: Discuss the basic structure of a C program.
Answer: The basic structure of a C program is given below:

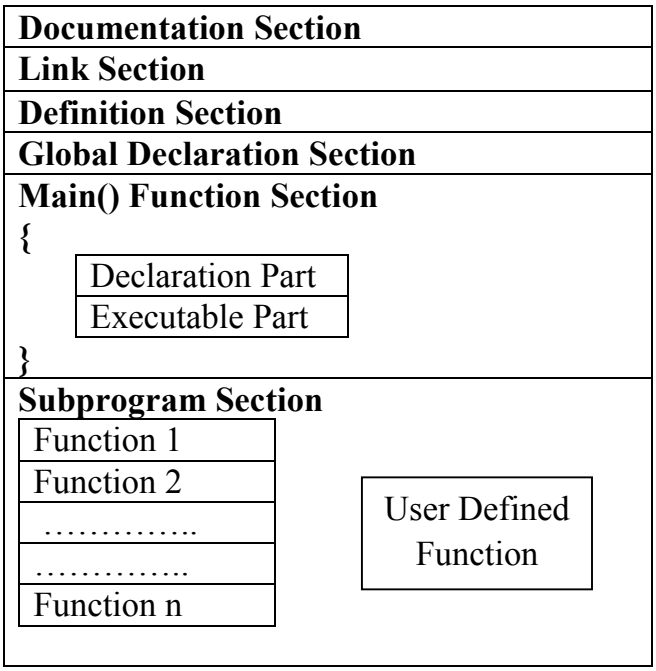


Figure: Basic Structure of a C program.

The documentation section consists of a set of comment lines giving the name of the program, the author, and other details which the programmer would like to use later.

The link section provides instructions to the compiler to link function from the system library.
The definition section defines all symbolic constants.

There are some variables which are used in more than one function. Such variables are called global variables and are declared in the **global declaration section** that is outside of all the function.
Every C program must have one **main () function** section. This section contains two parts: **declaration part & executable part**. **The declaration part** declares all the variables used in the executable part. There is at least one statement in the executable part. These two parts must appear between the opening and the closing braces. The program execution begins at the opening brace and ends at the closing brace. The closing brace of the main function section is the logical end of the program. All statements in the declaration and executable parts end with a semicolon (**;**).

The subprogram section contains all the user defined functions that are called in the main function. User defined functions are generally placed immediately after the **main function**, although they may appear in any order.

Question: Draw the process of compiling and running a C program & explain briefly.

Answer: Executing a program written in C involves a series of steps. These are:

1. Creating the program,
2. Compiling the program,
3. Linking the program with functions that are needed from the C library, and
4. Executing the program.

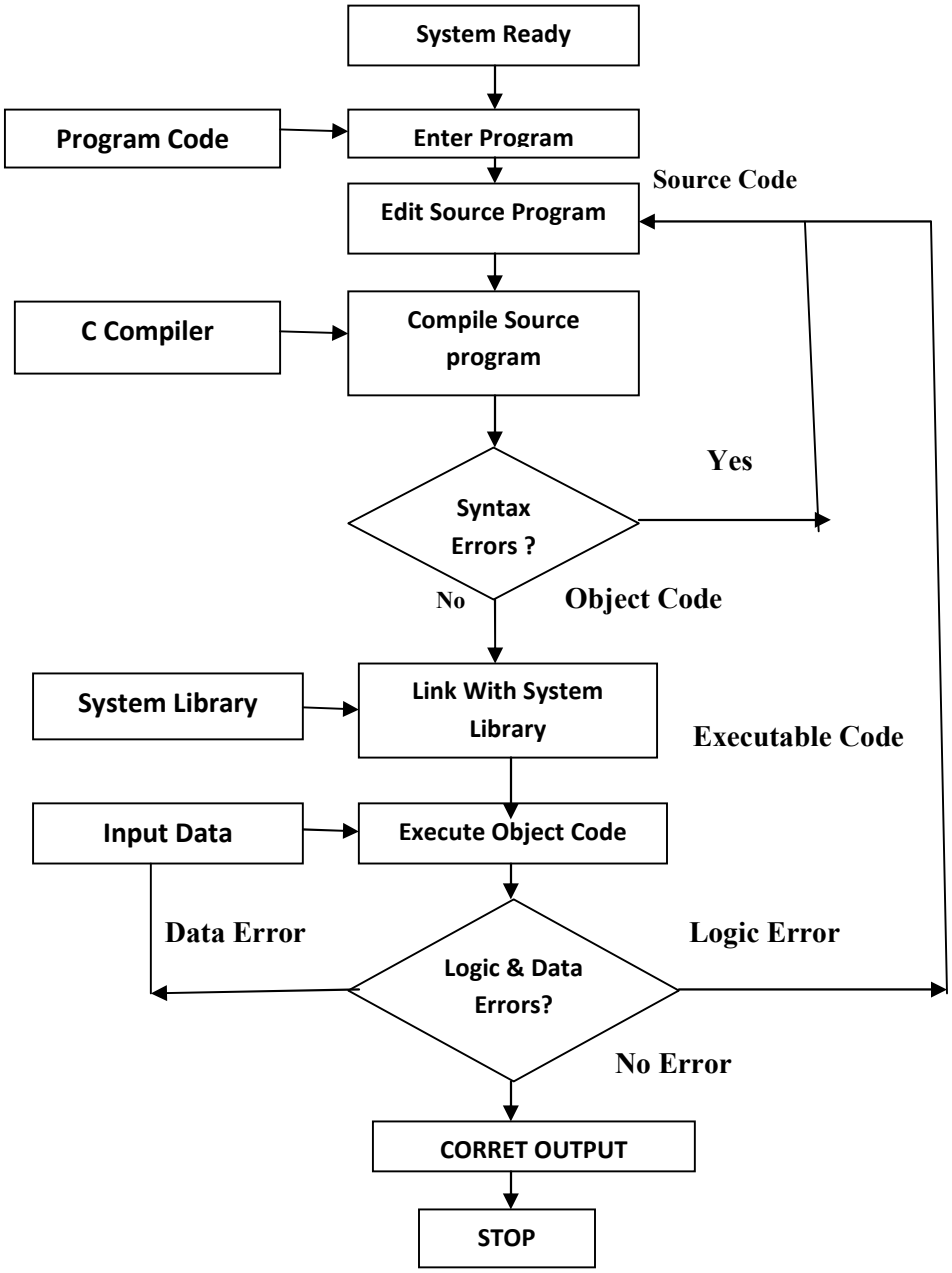


Figure: Process of compiling and running a C program.

At first, we should ready the system, and then we will enter the program in the system which is called the source code.
Then we can edit the source program when it is needed. After edit, we can compile the source program and check it, where contains the syntax error or not? If it has the syntax error, then we should handle the error or correction the error.
After correction that error or if there is no error, we will create the link with the system library for executing the program and translates the source program to object program.

Then we will check the logic error and data error, when the data error is find out, it requires to correction for input data and send to the executable object code. Lastly, if it has no error, it will show the correct output and stop the program.

Besides, the logic error is find out, it will send the for correcting error to the edit source program and execute step by step every statement sequentially. Finally, it will show the correct results and stop the programs.

Question: Write the short note of following terms for C programs:

I. Creating,
II. Compiling,
III. Linking,
IV. Executing.

Answer:

Creating: Creating is the process to create the programs. When the system is ready, then the program must be entered into a file. The file name can consist of letters, digits, and special characters, followed by a dot and a letter c. Example: add.c, sub.c, mul.c, div.c etc.

The file is created with the help of a text editor.

Compiling: Compiling is the process to compile the program from source code to object code & checking the error and handling this error.

Linking: Linking refers to the creation of a single executable file from multiple object files.

Executing: Executing is the process to execute the program for getting the results or output.