

## Structure of the C Program

The basic structure of a C program is divided into 6 parts which makes it easy to read, modify, document, and understand in a particular format. C program must follow the below-mentioned outline in order to successfully compile and execute. Debugging is easier in a well-structured C program.

### Sections of the C Program

There are 6 basic sections responsible for the proper execution of a program. Sections are mentioned below:

1. Documentation
2. Preprocessor Section
3. Definition
4. Global Declaration
5. Main() Function
6. Sub Programs

#### 1. Documentation

This section consists of the description of the program, the name of the program, and the creation date and time of the program. It is specified at the start of the program in the form of comments. Documentation can be represented as:

```
// description, name of the program, programmer name, date, time etc.
```

**or**

```
/*  
    description, name of the program, programmer name, date, time etc.  
*/
```

Anything written as comments will be treated as documentation of the program and this will not interfere with the given code. Basically, it gives an overview to the reader of the program.

#### 2. Preprocessor Section

All the header files of the program will be declared in the [preprocessor](#) section of the program. Header files help us to access other's improved code into our code. A copy of these multiple files is inserted into our program before the process of compilation.

**Example:**

```
#include<stdio.h>
#include<math.h>
```

**3. Definition**

Preprocessors are the programs that process our source code before the process of compilation. There are multiple steps which are involved in the writing and execution of the program. Preprocessor directives start with the '#' symbol. The #define preprocessor is used to create a constant throughout the program. Whenever this name is encountered by the compiler, it is replaced by the actual piece of defined code.

**Example:**

```
#define long long ll
```

**4. Global Declaration**

The global declaration section contains global variables, function declaration, and static variables. Variables and functions which are declared in this scope can be used anywhere in the program.

**Example:**

```
int num = 18;
```

**5. Main() Function**

Every C program must have a main function. The main() function of the program is written in this section. Operations like declaration and execution are performed inside the curly braces of the main program. The return type of the main() function can be int as well as void too. void() main tells the compiler that the program will not return any value. The int main() tells the compiler that the program will return an integer value.

**Example:**

```
void main()
```

**or**

```
int main()
```

**6. Sub Programs**

User-defined functions are called in this section of the program. The control of the program is shifted to the called function whenever they are called from the

main or outside the main() function. These are specified as per the requirements of the programmer.

**Example:**

```
int sum(int x, int y)
{
    return x+y;
}
```

**Structure of C Program with example**

**Example:** Below C program to find the sum of 2 numbers:

*// Documentation*

*/\*\**

*\* file: sum.c*

*\* author: you*

*\* description: program to find sum.*

*\*/*

*// Link*

*#include <stdio.h>*

*// Definition*

*#define X 20*

*// Global Declaration*

*int sum(int y);*

*// Main() Function*

*int main(void)*

*{*

*int y = 55;*

```
printf("Sum: %d", sum(y));
return 0;
}
```

*// Subprogram*

```
int sum(int y)
{
    return y + X;
}
```

## Output

Sum: 75

## Explanation of the above Program

Below is the explanation of the above program. With a description explaining the program's meaning and use.

Sections	Description
<pre>/**  * file: sum.c  * author: you  * description:  program to find sum.  */</pre>	<p>It is the comment section and is part of the description section of the code.</p>
<pre>#include&lt;stdio.h&gt;</pre>	<p>Header file which is used for standard input-output. This is the preprocessor section.</p>
<pre>#define X 20</pre>	<p>This is the definition section. It allows the use of constant X in the code.</p>

Sections	Description
<b>int sum(int y)</b>	This is the Global declaration section includes the function declaration that can be used anywhere in the program.
<b>int main()</b>	main() is the first function that is executed in the C program.
<b>{...}</b>	These curly braces mark the beginning and end of the main function.
<b>printf("Sum: %d", sum(y));</b>	printf() function is used to print the sum on the screen.
<b>return 0;</b>	We have used int as the return type so we have to return 0 which states that the given program is free from the error and it can be exited successfully.
<b>int sum(int y)</b> <b>{</b> <b>    return y + X;</b> <b>}</b>	This is the subprogram section. It includes the user-defined functions that are called in the main() function.

### Steps involved in the Compilation and execution of a C program:

- Program Creation
- Compilation of the program
- Execution of the program
- The output of the program