

## **Facts about C programming Language**

- In 1988, the American National Standards Institute (ANSI) had formalized the C language.
- C was invented to write UNIX operating system.
- C is a successor of 'Basic Combined Programming Language' (BCPL) called B language.
- Linux OS, PHP, and MySQL are written in C.
- C has been written in assembly language.

## **Features of C programming Language**

- C is a robust language with a rich set of built-in functions and operators.
- Programs written in C are efficient and fast.
- C is highly portable; programs once written in C can be run on other machines with minor or no modification.
- C is a collection of C library functions; we can also create our function and add it to the C library.
- C is easily extensible.

## **Advantages of C Language**

- C is the building block for many other programming languages.
- Programs written in C are highly portable.
- Several standard functions are there (like in-built) that can be used to develop programs.
- C programs are collections of C library functions, and it's also easy to add functions to the C library.
- The modular structure makes code debugging, maintenance, and testing easier.

## **Disadvantages of C Language**

- C does not provide Object Oriented Programming (OOP) concepts.
- There are no concepts of Namespace in C.
- C does not provide binding or wrapping up of data in a single unit.
- C does not provide Constructor and Destructor.

## Syntax and Semantics

- The *syntax rules* of a language define how we can put together symbols, reserved words, and identifiers to make a valid program
- The *semantics* of a program statement define what that statement means (its purpose or role in a program)
- A program that is syntactically correct is not necessarily logically (semantically) correct
- A program will always do what we tell it to do, not what we meant to tell it to do

# Errors

- ✓ A program can have three types of errors
- ✓ The compiler will find syntax errors and other basic problems (*compile-time errors*)
  - ✓ If compile-time errors exist, an executable version of the program is not created
- ✓ A problem can occur during program execution, such as trying to divide by zero, which causes a program to terminate abnormally (*run-time errors*)
- ✓ A program may run, but produce incorrect results, perhaps using an incorrect formula (*logical errors*)

# C Basic Syntax

## Tokens in C

A token is either a keyword, an identifier, a constant, a string literal, or a symbol. For example, the following C statement consists of five tokens –

```
printf("Hello, World! \n");
```

The individual tokens are

```
printf  
(  
    "Hello, World! \n"  
)  
;
```



# C Basic Syntax

## Semicolon

In a C program, the semicolon is a statement terminator. That is, each individual statement must be ended with a semicolon. It indicates the end of one logical entity.

```
printf("Hello, World! \n");  
return 0;
```

## Comments

Comments are like helping text in your C program and they are ignored by the compiler. Single Line comment start with `//` and multiline comment start with `/*` and terminate with the characters `*/` as shown below

`// Single line comment`

`/* Demonstration of`

`Multiline comment */`

# C Basic Syntax

## Identifier

A C identifier is a name used to identify a variable, function, or any other user-defined item. An identifier starts with a letter A to Z, a to z, or an underscore '\_' followed by zero or more letters, underscores, and digits (0 to 9).

C does not allow punctuation characters such as @, \$, and % within identifiers.

C is a case-sensitive programming language.

Thus, **Manpower** and **manpower** are two different identifiers in C. Here are some examples of acceptable identifiers

mohd	zara	abc	move_name	a_123
myname50	_temp	j	a23b9	retVal

# C Basic Syntax

## Keywords

Keywords are the reserved words in C. These reserved words may not be used as constants or variables or any other identifier names. List of 32 keywords are:

auto	double	int	struct
break	else	long	switch
case	enum	register	typedef
char	extern	return	union
const	float	short	unsigned
continue	for	signed	void
default	goto	sizeof	volatile
do	if	static	while

## Whitespace

Whitespace is the term used in C to describe blanks, tabs, newline characters and comments. Whitespace separates one part of a statement from another.

---



# C Data Type

C provides various types of data-types which allow the programmer to select the appropriate type for the variable to set its value.

The data-type in a programming language is the collection of data with values having fixed meaning as well as characteristics. Some of them are an integer, floating point, character, etc. Usually, programming languages specify the range values for given data-type.

**ANSI C provides three types of data types:**

## **1.Primary(Built-in) Data Types:**

*void, int, char, double and float.*

## **2.Derived Data Types:**

*Array, References, and Pointers.*

## **3.User Defined Data Types:**

*Structure, Union, and Enumeration.*

# C Data Type

## 1.Primary(Built-in) Data Types:

void	As the name suggests, it holds no value and is generally used for specifying the type of function or what it returns. If the function has a void type, it means that the function will not return any value.
int	Used to denote an integer type.
char	Used to denote a character type.
float, double	Used to denote a floating point type.
int *, float *, char *	Used to denote a pointer type.

## Declaration

```
int    age;  
char   letter;  
float  height, width;
```

# C Data Type

## Integer Data Types:

Type	Storage size	Value range
char	1 byte	-128 to 127
unsigned char	1 byte	0 to 255
int	2 or 4 bytes	-32,768 to 32,767 or -2,147,483,648 to 2,147,483,647
unsigned int	2 or 4 bytes	0 to 65,535 or 0 to 4,294,967,295
short	2 bytes	-32,768 to 32,767
unsigned short	2 bytes	0 to 65,535
long	8 bytes	-9223372036854775808 to 9223372036854775807
unsigned long	8 bytes	0 to 18446744073709551615

# C Data Type

**Floating point Data Types:**

Type	Storage size	Value range	Precision
float	4 byte	1.2E-38 to 3.4E+38	6 decimal places
double	8 byte	2.3E-308 to 1.7E+308	15 decimal places
long double	10 byte	3.4E-4932 to 1.1E+4932	19 decimal places

## Data Type and Variable declaration in C

```
#include <stdio.h>
int main()
{
    int a = 4000; // positive integer data type
    float b = 5.2324; // float data type
    char c = 'Z'; // char data type
    long d = 41657; // long positive integer data type
    long e = -21556; // long -ve integer data type
    int f = -185; // -ve integer data type
    short g = 130; // short +ve integer data type
    short h = -130; // short -ve integer data type
    double i = 4.1234567890; // double float data type
    float j = -3.55; // float data type
}
```

## Sizeof Operator

The storage representation and machine instructions differ from machine to machine. sizeof operator can use to get the exact size of a type or a variable on a particular platform.

```
#include <stdio.h>
#include <limits.h>
int main()
{
    printf("Storage size for int is: %d \n", sizeof(int));
    printf("Storage size for char is: %d \n", sizeof(char));
    return 0;
}
```



## Derived Data Types

C supports 3 derived data types:

Data Types	Description
Arrays	Arrays are sequences of data items having homogeneous values. They have adjacent memory locations to store values.
References	Function pointers allow referencing functions with a particular signature.
Pointers	These are powerful C features which are used to access the memory and deal with their addresses.

## User defined Data Types

C allows the feature called type definition which allows programmers to define their identifier that would represent an existing data type. There are three such types:

Data Types	Description
Structure	It is a package of variables of different types under a single name. This is done to handle data efficiently. "struct" keyword is used to define a structure.
Union	These allow storing various data types in the same memory location. Programmers can define a union with different members, but only a single member can contain a value at a given time. It is used for
Enum	Enumeration is a special data type that consists of integral constants, and each of them is assigned with a specific name. "enum" keyword is used to define the enumerated data type.

**THANK YOU**