



PROGRAMMING AND HOW PROGRAM WORKS

Basic Concepts to Programming

COMPUTER PROGRAM

The computer is an ignorant machine that cannot do anything unless we tell it how to do it step by step. These steps are the programs or software. However, before writing a program a software developer needs to have a clear idea of which steps the program is to follow, so he makes an algorithm.



WHAT IS PROGRAMMING?

- Programming is the instruction to system for any kind of computer operation.



WHAT IS COMPUTER?

- Computer is an electronic device
- Store, manipulate and Transmit data
- Data can be
 - Numeric
 - Character (Name, Address etc)
 - Graphics (Charts, Photographs, Videos etc)
 - Sound (Music, Speech etc)



MEMORY

- Every information is stored in binary code.
- Binary code is the sequence of 0 and 1.
- 0 and 1 are called bits.
- 0 represents off and 1 represents ON.
- 8 bits makes 1 byte.
- Character takes one byte to store.
- Bit is the smallest unit of information storage.



PROGRAMMING LANGUAGE CONCEPT

- A programming language is a formal constructed language designed to communicate instruction to a machine particularly a computer. Program language can be used to create programs to control machine language. A programming languages is a special language programmers use to develop application, scripts, or other set of instructions for computers to execute .

TYPE OF PROGRAMMING LANGUAGES FOR DIFFERENT PURPOSE.

- To develop application and programs
 - C
 - C++
 - Java
 - Visual Basic
- To develop Database
 - Dbase
 - Foxpro
 - Oracle
 - My SQL

CONT...

- To develop Game
 - C
 - C#
 - Dark Basic
 - Java
- To develop internet and web page
 - HTML
 - Java Script
 - PHP
 - XML

CONT....

- To develop scripts
 - Autohotkey
 - Batchfile
 - Python
 - TCL
- To develop computer driver or hardware
 - Assembly
 - C

PROGRAMMING LANGUAGE

Computer is a machine which does not understand human language like English, Spanish, Nepali etc. it only understand machine language which is very complex to write and understand to us. Machine language is often referred to as a “low level language” since the code directly manipulates the hardware of computer.

While higher level languages such as “C”, C++, Pascal, Fortran etc are called compiled languages. In a compiled language, the programmer writes more general instructions and a compiler automatically translates these high level instruction into machine language.

PROGRAMMING LANGUAGE

- Machine Language
- Assembly Language
- High-Level Language



TRANSLATOR

Interpreter: It translate higher level language to machine language line by line to execute program. If any line has got error than program will terminate until fix that error.

Compiler: It translate higher level language to machine language to execute program. Its difference with interpreter because interpreter translate line by line but it translate whole program immediately if error not found.

HISTORY OF C

C is a programming language developed at AT & T's Bell laboratories of USA in 1972. IT was design and written by a man named Dennis Ritchie. In the late seventies C began to replace the more familiar languages of that time like PL/I, Algol, Fortran etc.

Possibly why C seems to popular is because it is reliable, simple and easy to use.

WHY C?

- Nowadays widely used programming language is C++, Java etc. But we can say nobody can learn these language directly. Because many features of these language is developed by C programming.
- C++, C#, Java make use of Object Oriented Programming (OOP) to organize the program. This principle has lot of advantages but to run this principle you would still need a elements of C.
- Major Part of operating system like Windows, Unix, Linux is still write in C. Because even today when it comes to perform (speed of execution) nothing beats C.

- Mobile device like cellular phone and palmtop are become increase popularly. And consumer device like Washing machine, TV remote, Micro Oven etc are use Micro processor also called chip. These type of chip also use program which is written in C.
- We have seen many professional computer game. These game are develop in graphical software. But to instruct the games or programs it use C.
- The increasing of hardware device day by day. To introduce these hardware device computer needs driver to know what the purpose of these device. Hence the device driver is also written by C.

GETTING STARTED WITH C

- C is a compiling programming language which translate higher level language to machine level language to execute programs. So to execute program written in C we need compiler. We can write program code any where like ms word, notepad, excel etc but it does not execute. So we need compiler like Dev C++, Turbo C, C free etc.

PROGRAM STRUCTURE

Virtually all structure programs share a similar overall pattern:

- Statements to establish the start of the program.
- Variable declaration.
- Program statements(blocks of code).

The following is a simple program written in C which give us output text as “Welcome to AITE”.

```
#include<stdio.h>
Void main()
{
Printf(“Welcome to AITE”);
}
```

RULES THAT APPLICABLE TO ALL C PROGRAMS

- Each instruction in a C program is written as a separate statement. Therefore a complete C program would comprise of a **series of statements**.
- Blank spaces may be inserted between two words to improve the readability of the statements. however, **no blank spaces are allowed** within a variable, constant or keyword.
- All statements are entered in **small case letters**.
- C has **no specific rules for the position** at which a statement is to be written, That's why it is often called a free-form language.
- Every **C statement must end with semicolon(;) which acts as statement terminator**.

C KEYWORDS

- Keywords are the words whose meaning has already been explained to the C compiler. The keywords cannot be used as variable names because if we do so we are trying to assign a new meaning to the keyword, which is not allowed.

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

VARIABLE DECLARATION

Variables are place holders for data a program might use or manipulate. Variables are given names so that we can assign values to them and refer to them later to read the values. Variables typically store values of a given types .

Types generally include:

Integer: To store integer or “whole” numbers.

Real: To store real numbers (also called float to indicate a floating point number)

Character: A single characters such as letter of the alphabet.

String: A collection of characters.

The compiler needs to know in advance the type of data that will be stored in it. For this reason, we declare the variables at the start of a program. Variable declaration consists of giving a new name and a data type for the variable.

In the following example programs, variables of different types are declared and used in the programs.

```
#include<stdio.h>
void main()
{
int age;
float salary;
char gender;
age=25;
salary=12500.80;
gender='M';
printf("Age= %d",age);
printf("Salary= %f", salary);
printf("Gender= %c", gender);
}
```

In the above program we declare three types of variable integer, float(real) and character. In program integer is declare as int, float as float and character as char. In program to print out integer it use %d.

While in above program age declare as variable which can store integer value. So it can store whole numbers only it can't store fractional or decimal value. In program to print out float it use %f.

Again, salary is declare as variable which can store float value. So it can store only fractional or decimal value.

At last, Gender is declare as character which can store a character value which must be in single quotes for eg 'A'. It is a character variable so it can store only 1 character in 1 variables. In program to print out character it use %c.

```
/* calculating total price */
#include<stdio.h>
#include<conio.h>
main()
{
int qty;
float rate, total;
printf(" Item's Quantity:");
scanf("%d", &qty);
printf("\nItem's rate:");
scanf("%f", &rate);
total=rate*qty;
printf("The total price of item is Rs. %f", total);
getch();
}
```

After execution of this program it asked input values like this

Item's quantity: 1000

Item's rate: 4.5

If we input value as above 1000 in quantity and 4.5 in rate the output of this program will be

The total price of item is Rs. 4500.00

When compiling above program let's look at what each line is doing.

Line1: This is one way to write comments in C, between /* and */ in program this is used to indicate the purpose of program.

Line2: The #include command tells compiler to look at source which it gives between < and >. Stdio.h stands for standard input and output and .h stands for header. We use printf() function for output and scanf() function for input. Does this function lies in <stdio.h> header. To use this function we must include this header.

Line3: As in line2 we include <conio.h> which stands for console input and output. We use getch() function in this program which lies in <conio.h> header.

Line4: The C program always starts with main() function.

Line5: Open braces { it indicate the start of program.

Line6: Declaring price as integer variables.

Line7: Declaring rate and total as float variables. We can declare many variable in single line by separating with comma.

- Line8: Using printf function to output text Item's Quantity:
- Line9: Using Scanf function to input value for Quantity. We use %d which indicate integer value. We use &(ampersand) which is used to know address of variables.
- Line10: Using printf function to output text Item's rate: . In this line we use\n in front of Item's rate. \n is use to line break.
- Line11: Using scanf function to input value for rate. We use %f which indicate float value.
- Line12: Calculating the total value by using arithmetic operation.
- Line13: Printing out the result of total value.
- Line14: Using getch() function which is use to catch display screen of result. Generally the program execute so fast we cant see result.
- Line15: Closing the braces } which indicate end of program.

COMMON PROGRAMMING CONCEPTS IN C

Function: A function is a block of code representing something the computer should do when the program runs. We can also create functions in C.

Variables: Sometimes you need the flexibility to run the program without knowing what type of values. C allows you to use variables when you need that flexibility like variables in algebra.

Data types: In order to store data in memory while your program is running, and to know what operations you can perform on that data, a programming language like C defines certain data types it will recognize. Each data type in C has a certain size, measured in binary bits or bytes, and a certain set of rules about what its bits represent.

Operation: In C, you can perform arithmetic operations (such as addition) on numbers and string (such as concatenation) on strings of characters. C also has built in operations specifically designed for things you might want to do with your data.

CONTD

Loops: One of the most basic things a programmer will want to do is repeat an action some number of times based on certain conditions that come up while the program is running. A block of code designed to repeat based on given conditions is called loop, and the C language provides for these common loop structures: while, do/while, for, continue, break and goto. C also include common if/then/else conditionals and switch/case statements.

Data structures: When your program has a lot of data to handle, and you need to sort search through that data, you'll probably use some sort of data structure. A data Structure is a structured way of representing several pieces of data of the same data type.

Preprocessor operations: Sometimes you'll want to give the compiler some instructions on things to do with your code before compiling it into the executable. These operations include substituting constant values and including code from C libraries(which you saw in the sample code earlier.)