CCA-101: Fundamentals of IT & Programming

Assignment – 2

Q1. What is the difference between Machine Language and High Level Language?

Ans.1. A **high-level language** is a programming language that uses English and mathematical symbols, like +, -, % and many others, in its instructions. When using the term 'programming languages,' most people are actually referring to high-level languages. High-level languages are the languages most often used by programmers to write programs. Examples of high-level languages are C++, Fortran, Java and Python.

2. **Machine language**, or **machine code**, is the only language that is directly understood by the computer, and it does not need to be translated. All instructions use binary notation and are written as a string of 1s and 0s. A program instruction in machine language may look something like this:

```
1.10010101100101001111101010011011100101
```

Q2. Discuss about different data types of C programming Language.

Ans. Data types in C Language

Data types specify how we enter data into our programs and what type of data we enter. C language has some predefined set of data types to handle various kinds of data that we can use in our program. These datatypes have different storage capacities.

C language supports 2 different type of data types:

1. **Primary data types**:

These are fundamental data types in C namely integer(`int`), floating point(`float`), character(`char`) and `void`.

2. **Derived data types**:

Derived data types are nothing but primary datatypes but a little twisted or grouped together
like **array**, **stucture**, **union** and **pointer**. These are discussed in details later.

Data type determines the type of data a variable will hold. If a variable `x` is declared as `int`. it means x can hold only integer values. Every variable which is used in the program must be declared as what data-type it is.

Integer type

Integers are used to store whole numbers.

Size and range of Integer type on 16-bit machine:

Type	Size(bytes)	Range
int or signed int	2	-32,768 to 32767

unsigned int	2	0 to 65535
short int or signed short int	1	-128 to 127
unsigned short int	1	0 to 255
long int or signed long int	4	-2,147,483,648 to 2,147,483,64
unsigned long int	4	0 to 4,294,967,295

Floating point type

Floating types are used to store real numbers.

Size and range of Integer type on 16-bit machine

Type	Size(bytes)	Range
Float	4	3.4E-38 to 3.4E+38
double	8	1.7E-308 to 1.7E+308

long double	10	3.4E-4932 to 1.1E+4932

Character type

Character types are used to store characters value.

Size and range of Integer type on 16-bit machine

Type	Size(bytes)	Range
char or signed char	1	-128 to 127
unsigned char	1	0 to 255

---

void type

`void` type means no value. This is usually used to specify the type of functions which returns nothing. We will get acquainted to this datatype as we start learning more advanced topics in C language, like functions, pointers etc.

Q3. Find the output of the following expressions

a) X=20/5*2+30-5 b) Y=30 – (40/10+6) +10
c) Z= 40*2/10-2+10

ans

Q4. Describe the syntax of the following statements

a) If – else statement b) for loop c) while loop d) do-while loop

ans**.** In the last tutorial we learned how to use if statement in C. In this guide,

we will learn how to use if else, nested if else and else if statements in a C

Program.

C If else statement

Syntax of if else statement:

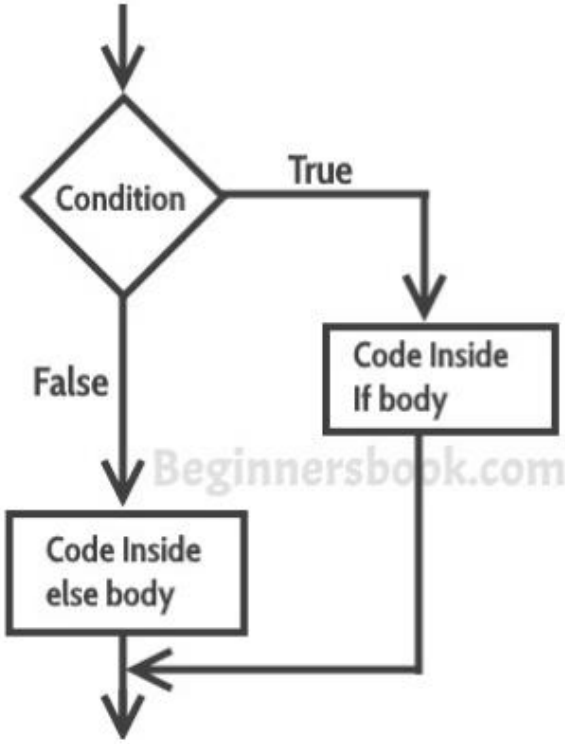
If condition returns true then the statements inside the body of “if” are

executed and the statements inside body of “else” are skipped.

If condition returns false then the statements inside the body of “if” are

skipped and the statements in “else” are executed.

```
if(condition) {
  // Statements inside body of if
}
else {
  //Statements inside body of else
```



Example of if else statement

In this program user is asked to enter the age and based on the input, the if..else statement checks whether the entered age is greater than or equal to 18. If this condition meet then display message “You are eligible for voting”, however if the

condition doesn't meet then display a different message "You are not eligible for voting".

b) for loop

Ans. What Is a For Loop?

A for loop enables a particular set of conditions to be executed repeatedly until a condition is satisfied. Imagine a situation where you would have to print numbers from 1 to 100. What would you do? Will you type in the printf command a hundred times or try to copy/paste it? This simple task would take an eternity. Using a for loop you can perform this action in three statements. This is the most basic example of the for loop. It can also be used in many advanced scenarios depending on the problem statement.

Check out the flowchart of a for loop to get a better idea of how it looks:

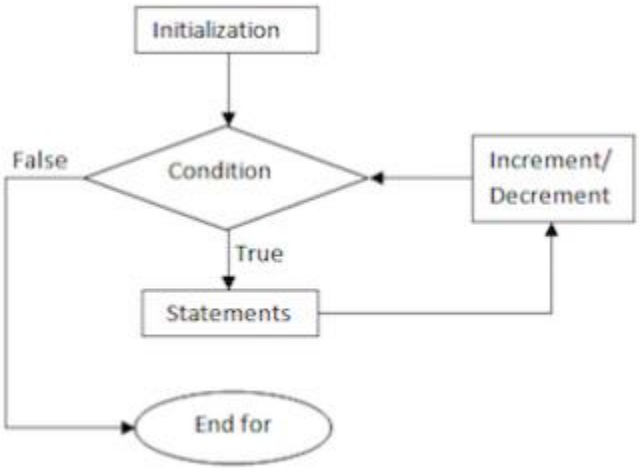


fig: Flowchart for for loop

Syntax of a For Loop

1. for (initialization statement; test expression;

update statement) {

2. // statements

3. }

c) while loop

Ans. A while loop in C programming repeatedly executes a target statement as long as a given condition is true.

Syntax

The syntax of a while loop in C programming language

is –

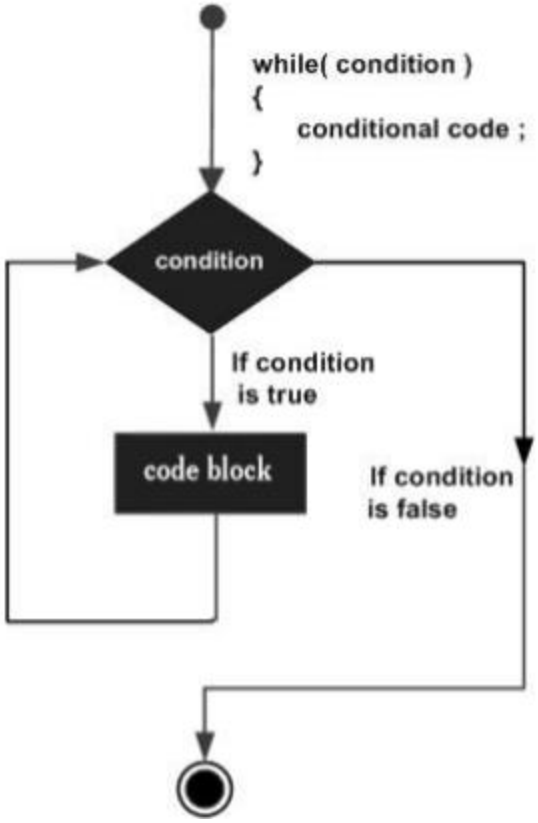
```
while(condition) {
statement(s);
```

Here, statement(s) may be a single statement or a block of statements.

The condition may be any expression, and true is any nonzero value. The loop iterates while the condition is true.

When the condition becomes false, the program control passes to the line immediately following the loop.

Flow Diagram



Here, the key point to note is that a while loop might not execute at all. When the

condition is tested and the result is false, the loop body will be skipped and the first

statement after the while loop will be executed.

Example

```c
#include <stdio.h>

int main () {

   /* local variable definition */
   int a = 10;

   /* while loop execution */
   while( a < 20 ) {
      printf("value of a: %d\n", a);
      a++;
   }

   return 0;
}
```

When the above code is compiled and executed, it produces the

following result −

value of a: 10

value of a: 11

value of a: 12

value of a: 13

value of a: 14

value of a: 15

value of a: 16

value of a: 17

value of a: 18

value of a: 19

d) do-while loop

Ans.

Unlike for and while loops, which test the loop condition at the top of the loop,

the do...while loop in C programming checks its condition at the bottom of the loop.

A do...while loop is similar to a while loop, except the fact that it is guaranteed to execute

at least one time.

Syntax

The syntax of a do...while loop in C programming language is −

do {

statement(s);

} while(condition);

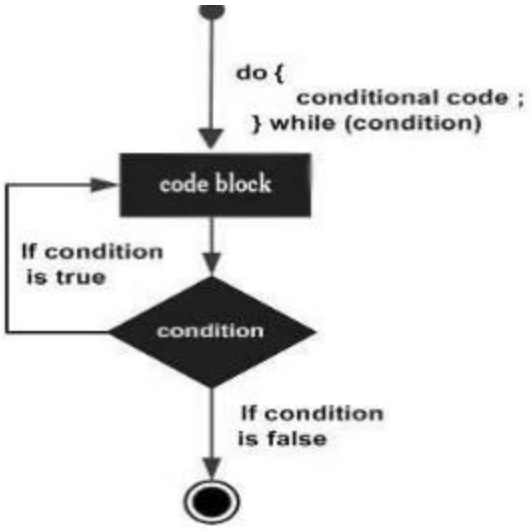
Notice that the conditional expression appears at the end of the loop, so the statement(s)

in the loop executes once before the condition is tested.

If the condition is true, the flow of control jumps back up to do, and the statement(s) in the

loop executes again. This process repeats until the given condition becomes false.

Flow Diagram



Example

```
#include <stdio.h>

int main () {

  /* local variable definition */
  int a = 10;

  /* do loop execution */
  do {
    printf("value of a: %d\n", a);
    a = a + 1;
  }while( a < 20 );

  return 0;
}
```

When the above code is compiled and executed, it produces the

following result −

value of a: 10

value of a: 11

value of a: 12

value of a: 13

value of a: 14

value of a: 15

value of a: 16

value of a: 17

value of a: 18

value of a: 19