Basic of C Programming

LOOP

Definition

- The looping can be defined as repeating the same process multiple times until a specific condition satisfies. There are three types of loops used in the C language. In this part of the tutorial, we are going to learn all the aspects of C loops.

- **Advantage of loops in C –**

1) It provides code reusability.
2) Using loops, we do not need to write the same code again and again.
3) Using loops, we can traverse over the elements of data structures (array or linked lists).

Types of C Loops

There are three types of loops in C language that is given below –

1) **do-while**
2) **while**
3) **for**

NOTE:

- **Entry Controlled Loop -** Loop, where test condition is checked before entering the loop body, known as Entry Controlled Loop.

  Example – while loop, for loop

- **Exit Controlled Loop -** Loop, where test condition is checked after executing the loop body, known as Exit Controlled Loop.

  Example – do – while loop

Entry Controlled Loop vs. Exit Controlled Loop

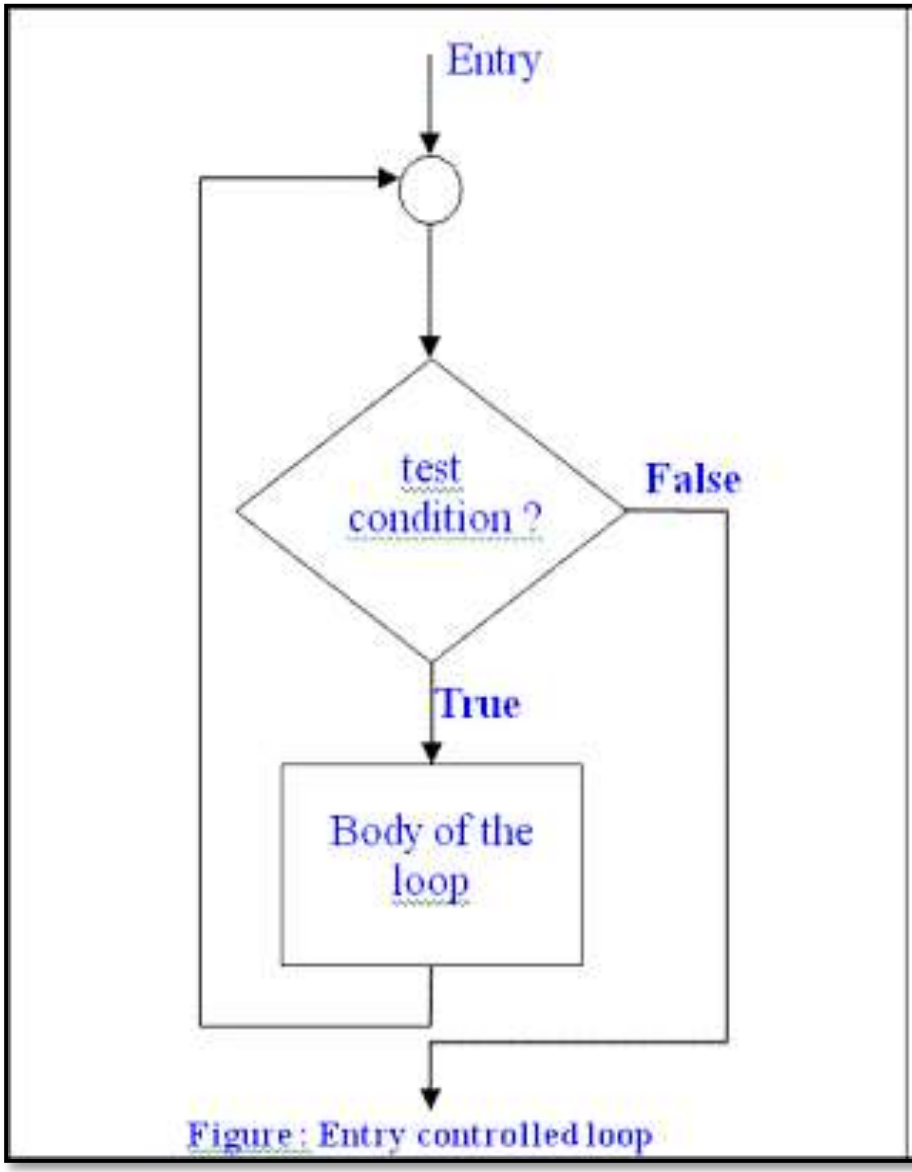

Figure : Entry controlled loop

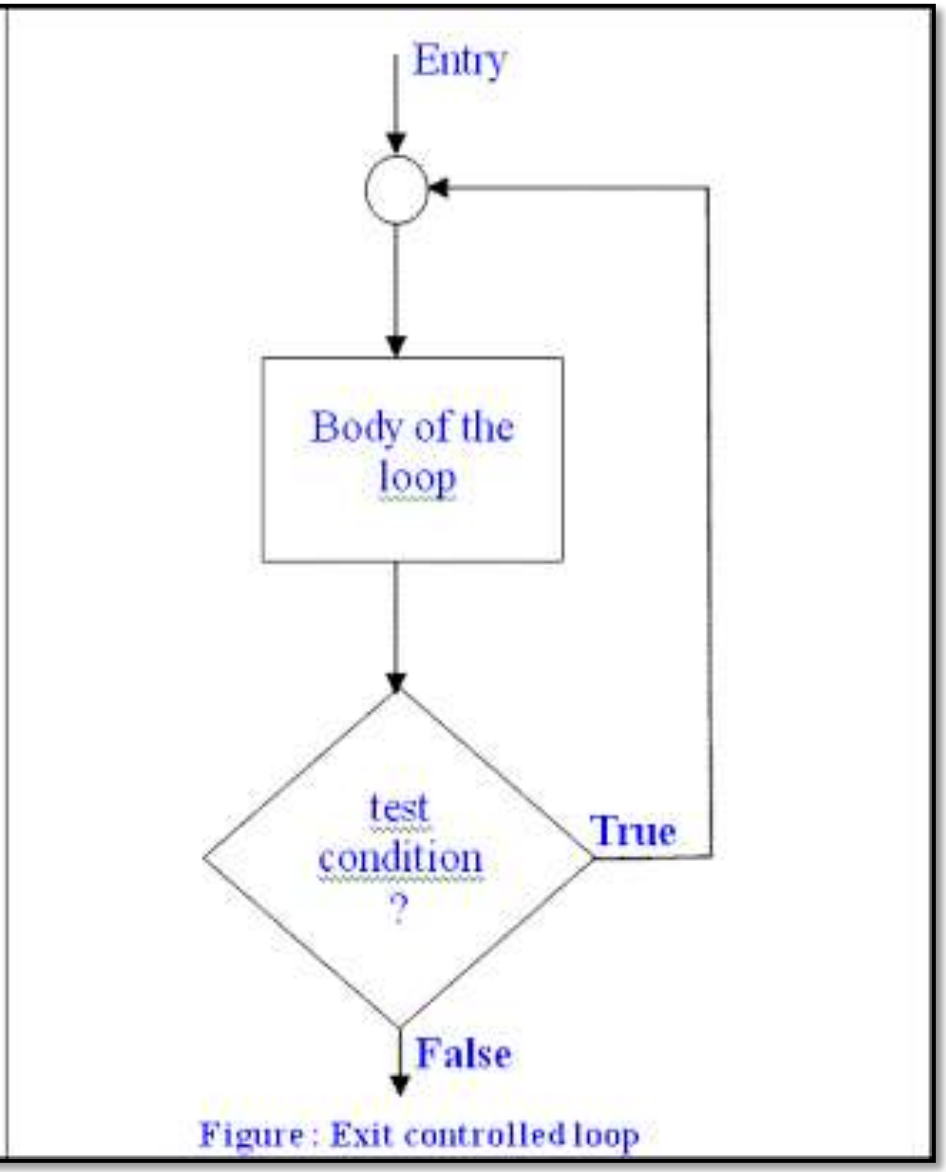

Figure : Exit controlled loop

Entry Controlled Loop	Exit Controlled Loop
Test condition is checked first, and then loop body will be executed.	Loop body will be executed first, and then condition is checked.
If Test condition is false, loop body will not be executed.	If Test condition is false, loop body will be executed once.
for loop and while loop are the examples of Entry Controlled Loop.	do while loop is the example of Exit controlled loop.
Entry Controlled Loops are used when checking of test condition is mandatory before executing loop body.	Exit Controlled Loop is used when checking of test condition is mandatory after executing the loop body.

while Loop

- The **while loop** in c is to be used in the scenario where we don't know the number of iterations in advance. The block of statements is executed in the while loop until the condition specified in the while loop is satisfied. It is also called a pre-tested loop.

- **Syntax –**

```
while(condition)
{
        //code to be executed
}
```

For Loop

- A **for loop** is a repetition control structure that allows you to efficiently write a loop that needs to execute a specific number of times.

- **Syntax –**

```
for (initialization; testExpression; update)
{
        // statements inside the body of loop
}
```

Do-while 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 –**

```
do
{
        //statement(s);
} while(condition);
```

Difference between while and do-while loop

- POST-TEST loop (exit-condition)
- The looping condition is tested after executing the loop body.
- Loop body is always executed at least once.

- PRE-TEST loop (entry-condition)
- The looping condition is tested before executing the loop body.
- Loop body may not be executed at all.