

Loops in C

Loops are used in programming to repeat a block of code until a specific condition is met. There are three loops in C programming:

1. for loop
2. while loop
3. do...while loop

For Loops

For loop is an entry controlled loop i.e. the condition is checked before entering into the loop. So if the condition is false for the first time, the statements inside while loop may not be executed at all. In order to exit from a for loop, either the condition should be false or a break statement should be encountered. For loop is suitable to use when we have to run a loop for a fixed number of times.

The basic format of for loop statement is:

Syntax:

```
for ( init; condition; increment )  
  
{  
  
    statement(s);  
  
}
```

Flowchart of for loop

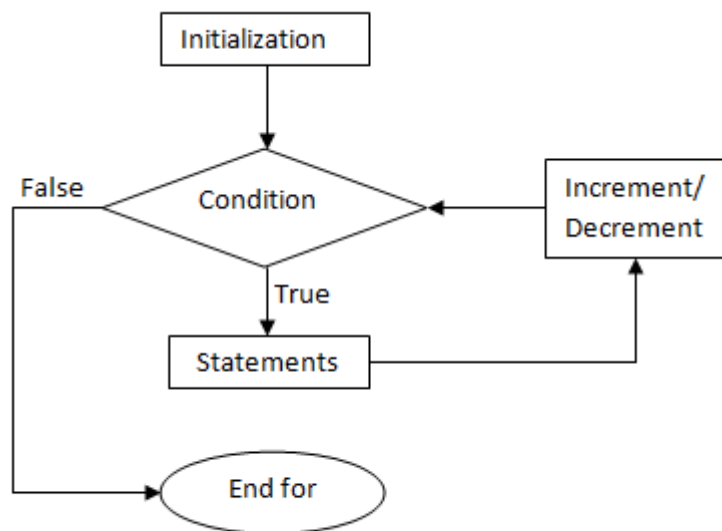


fig: Flowchart for for loop

While loop

While loop is an entry controlled loop i.e. the condition is checked before entering into the loop. So if the condition is false for the first time, the statements inside while loop may not be executed at all. The condition to be checked can be changed inside loop by changing values of variables. When the condition becomes false, the program control exits the loop. We can also exit a loop by using break statement like in switch case.

Syntax:

```
while (condition)
```

```
{
```

```
    statement(s);
```

```
    ... ..
```

```
}
```

Flowchart of while loop

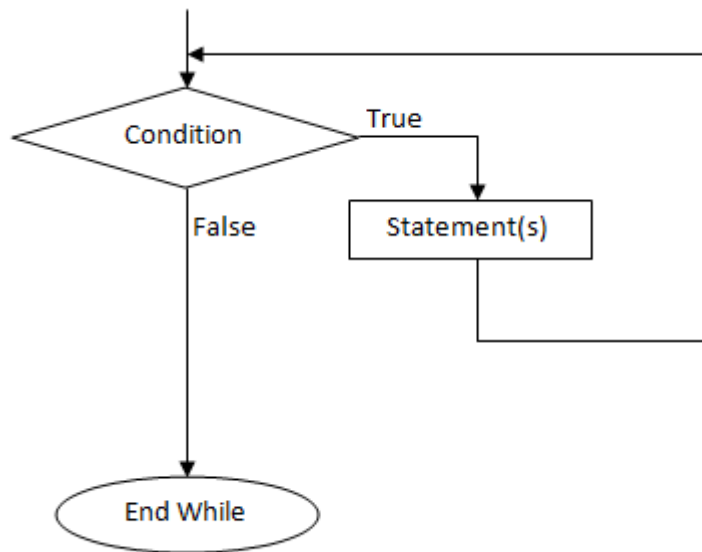


fig: Flowchart for while loop

Do-while loop

Do-while loop is an exit controlled loop i.e. the condition is checked at the end of loop. It means the statements inside do-while loop are executed at least once even if the condition is false. Do-while loop is an variant of while loop. In order to exit a do-while loop either the condition must be false or we should use break statement.

Syntax:

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

Flowchart of do-while loop

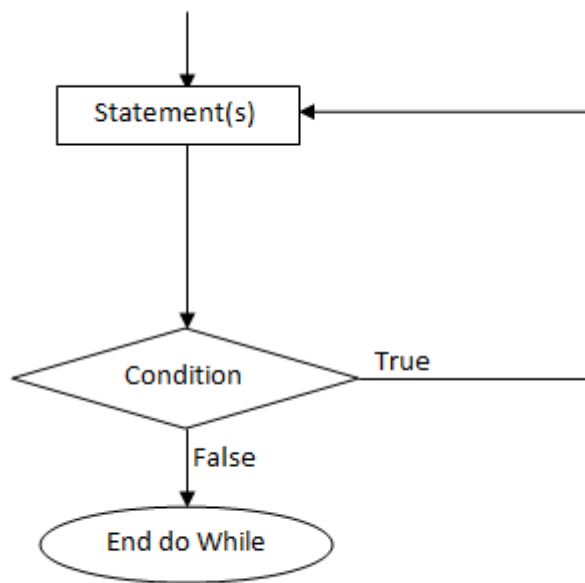


fig: Flowchart for do-while loop