BRAINWARE UNIVERSITY

Name : Anirban Kundu
Program Name : BCSE(AI&ML)
Course Code : SCM201
Course Name : Programing for problem solving
Student code : BWU/BTA/22/068

While loop vs do-while loop

CONTENT

- *WHAT IS LOOP*
- *TYPES OF LOOPS*
- *WHAT IS WHILE LOOP*
- *WHAT IS DO-WHILE LOOP*
- *EAMPLES*

What is loop?

In computer programming, a loop is a sequence of instruction s that is continually repeated until a certain condition is reached.
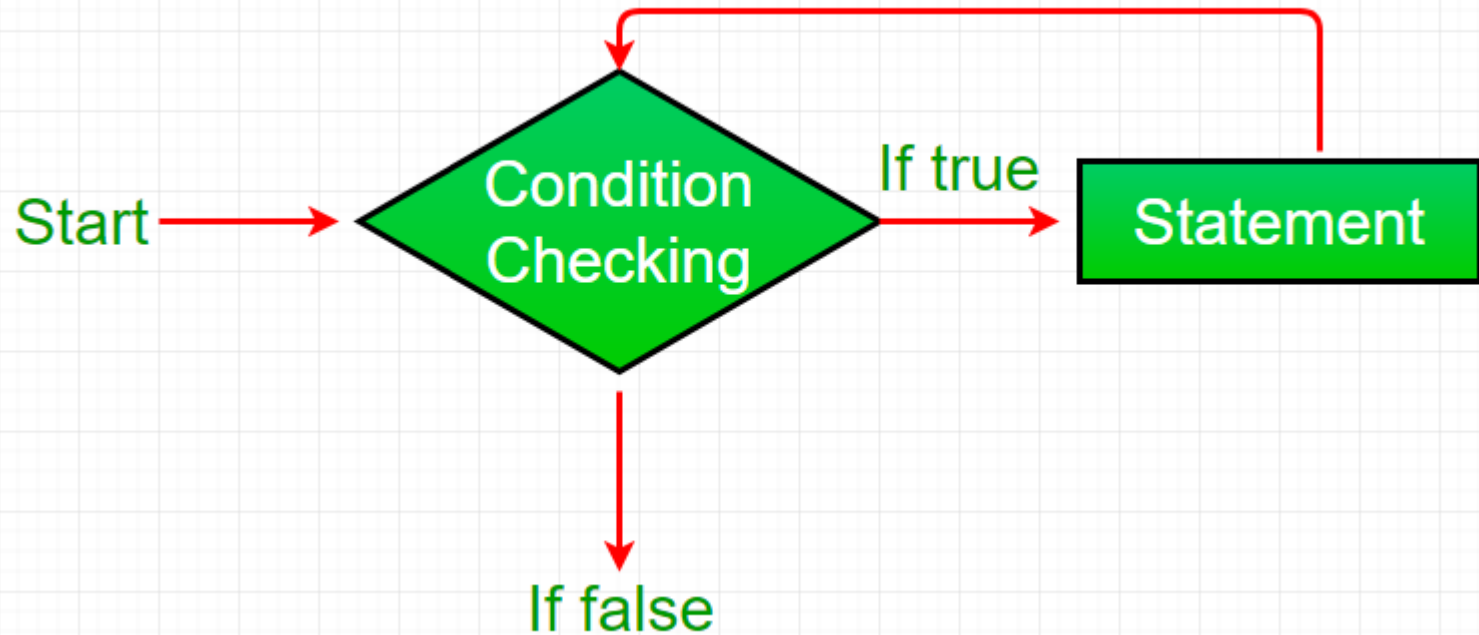
Types of loops…

In C programming, there are three loops:

- For Loop
- While Loop
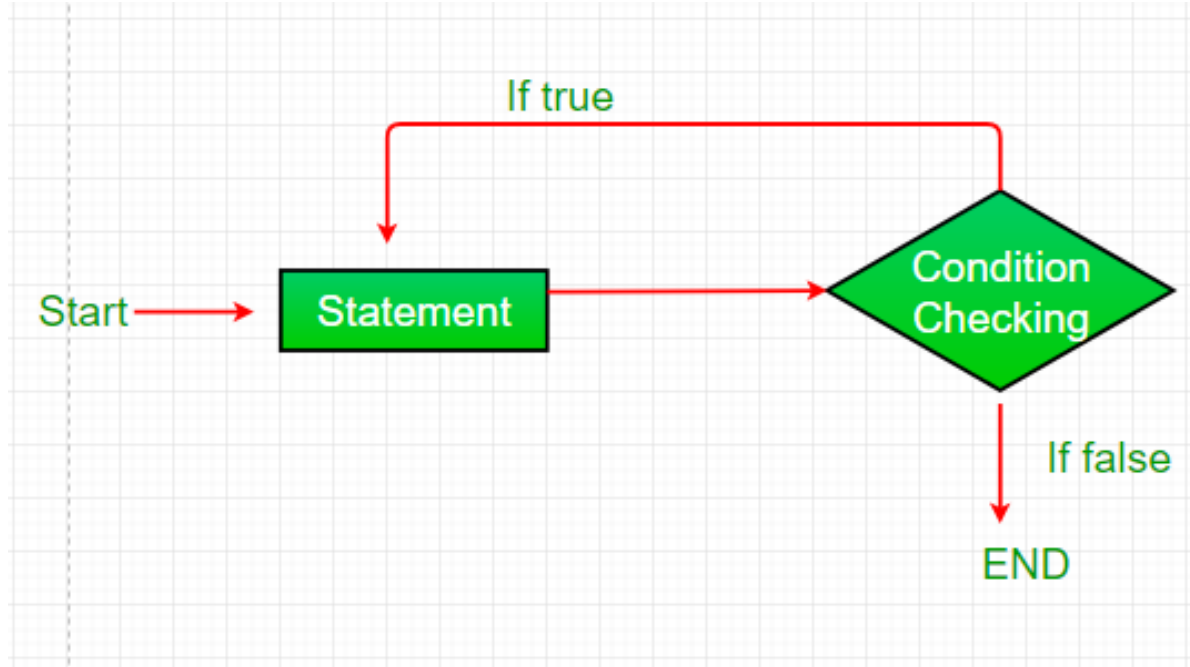- do-While Loop.

Flowchart While loop

What is while loop ?

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

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

Syntax of while loop

Flowchart Do-while loop

What is do-while loop?

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.

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

Syntax of do while loop

Different between while loop vs do while loop

WHILE	DO-WHILE
Condition is checked first then statement(s) is executed.	*Statement(s) is executed at least once, thereafter condition is checked.*
No semicolon at the end of while. *while(condition)*	*Semicolon at the end of while.* *while(condition);*

Different between while loop vs do while loop

WHILE	DO-WHILE
If there is a single statement, brackets are not required.	*Brackets are always required.*
while loop is entry controlled loop.	*do-while loop is exit controlled loop.*

EXAMPLE OF WHILE AND DO WHILE LOOP

```c
include <stdio.h>

int main() {
  int i = 0;

  while (i < 5) {
    printf("%d\n", i);
    i++;
  }

  return 0;
}
```

```c
include <stdio.h>

int main() {
  int i = 0;

  do {
    printf("%d\n", i);
    i++;
  } while (i < 5);

  return 0;
}
```

OUTPUT=>

```
0
1
2
3
4
```

```
0
1
2
3
4
```

THANK YOU