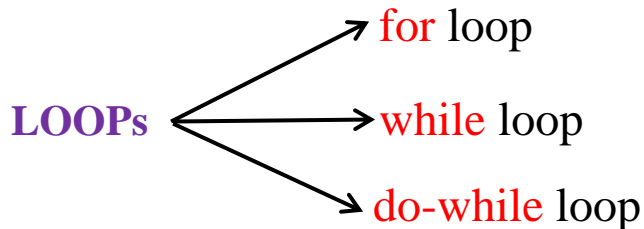


while Loop in C++

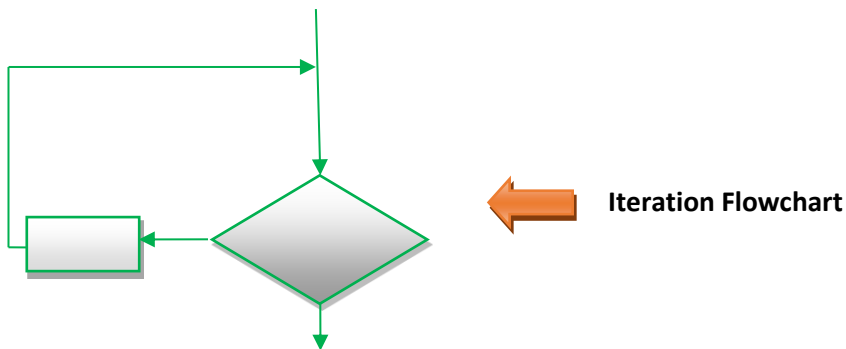
Loops programming construct is used to execute one or more instructions repeatedly until some condition is satisfied.

3 types of loops exist:



while LOOP

while loop also known as **Iteration**. Iteration logic is used when one or more instructions may be executed several times depending on some condition.



ITERATION

- Iteration comes from the word “reiterate”, which means to repeat
- Iteration is a looping construct
- Iteration is a combination of decision and sequence and can repeat steps
- Iteration can be thought of as “while something is true, do this, otherwise stop”

while loop syntax

initialization;

while (condition)

{

.....

.....

.....

.

.

.

.....

← program statements

increment or decrement;

}

The statements inside while loop gets executed until the condition become **FALSE**.

For loop also known as **iterative** statement. **To iterate means to repeat**. If we want to repeat execution of some action or statements several times, we use LOOPS.

Practice Programs

```
(i)
#include<iostream>
#include<cstdlib>
using namespace std;

int main()
{
    system("color fc");

    int x;

    x = 0;
    while(x<5)
    {
        cout<<"Stephen Hawking"<<endl;
        x++;
    }

    return 0;
}
```

```
(ii)
#include<iostream>
using namespace std;

int main()
{
    int x, y;

    x=1;
    while(y = 5)
    {
        cout<<"Stephen Hawking"<<endl;
        x++;
    }

    return 0;
}
```

```
(iii)
#include<iostream>
using namespace std;

int main()
{
    int x, y = 10;

    x=2;
    while(y == 10)
    {
        cout<<"Stephen Hawking"<<endl;
        x++;
    }

    return 0;
}
```

```
(iv)
#include<iostream>
using namespace std;

int main()
{
    int i, x, y;
    x = 2;
    y = 3;

    i=2;
    while(x+y)
    {
        cout<<"Stephen Hawking"<<endl;
        i++;
    }
    return 0;
}
```

```
(v)
#include<iostream>
using namespace std;

int main()
{
    int i, x, y;
    x = 2;
    y = 3;

    i=2;
    while((x+y)>0)
    {
        cout<<"Stephen Hawking"<<endl;
        i++;
    }

    return 0;
}
```

```
(vi)
#include<iostream>
using namespace std;

int main()
{
    int i, x, y;
    x = 2;
    y = 3;

    i=2;
    while((x+y)<0)
    {
        cout<<"Stephen Hawking"<<endl;
        i++;
    }

    return 0;
}
```

```

(vii)
#include<iostream>
using namespace std;

int main()
{
    int i;

    i=2;
    while(i<=5)
    {
        cout<<"Stephen Hawking"<<endl;
        cout<<"Galaxy"<<endl;
        i++;
    }
    return 0;
}

```

```

(viii)
#include<iostream>
using namespace std;

int main()
{
    int i, x, y;

    for(i = 2; i<=5; i++)

        cout<<"Stephen Hawking"<<endl;
        cout<<"Cosmologist"<<endl;
        cout<<"Galaxy"<<endl;

    return 0;
}

```

```

(ix)
#include<iostream>
using namespace std;

int main()
{
    int i, x, y;

    i=2;
    while(i<=5)
    {
        cout<<"Stephen Hawking"<<endl;
        cout<<"Cosmologist"<<endl;
        cout<<"Galaxy"<<endl;
        i++;
    }

    return 0;
}

```

Program explanations

Program 1

```
#include<iostream>
using namespace std;

int main()
{
    int x;
    x=0;
    while(x<5)
    {
        cout<<"Stephen Hawking"<<endl;
        x++;
    }
    return 0;
}
```

```
x = 0;
while(x<5)
{
    cout<<"Stephen Hawking"<<endl;
    x++;
}
```

Value of x	Condition x<5	Output
0	0<5 True	Stephen Hawking
1	1<5 True	Stephen Hawking
2	2<5 True	Stephen Hawking
3	3<5 True	Stephen Hawking
4	4<5 True	Stephen Hawking
5	5<5 False	<ul style="list-style-type: none">• Loop stops here as conditions becomes false• For loop come out of loop when condition becomes false• As long as condition is true, the statements inside for loop gets executed

Program 2

```
#include<iostream>
#include<cstdlib>
using namespace std;

int main()
{
    system("color fc");
    int x;

    x = 0;
    while(x<5)
    {
        cout<<"Stephen Hawking"<<endl;
        x++;
    }

    return 0;
}
```

Used to change **output window** background color and text color

Program 3

```
#include<iostream>
using namespace std;

int main()
{
    int x, y;
    x=1;
    while(y = 5)
    {
        cout<<"Stephen Hawking"<<endl;
        x++;
    }

    return 0;
}
```

x = 1;
while(y=5)
initial value of x = 1
y = 5 is condition
5 is put into y and 5 is non-zero. So, the condition is always true.
So gives infinite loop

Program 4

```
#include<iostream>
using namespace std;

int main()
{
    int x, y = 10;
    while (y == 10)
    {
        cout<<"Stephen Hawking"<<endl;
        x++;
    }

    return 0;
}
```

int x, y=10;
while (y == 10)
10 == 10 → after comparing LHS & RHS, this statement returns TRUE
So, the statement following for loop gets executed infinitely (∞)

Program 5

```
#include<iostream>
using namespace std;
```

```
int main()
```

```
{
    int i, x, y;
    x = 2;
    y = 3;
```

```
    i=2;
    while(x+y)
```

```
{
    cout<<"Stephen Hawking"<<endl;
    i++;
```

```
    }
    return 0;
```

```
}
```

x = 2;

y = 3;

i=2;

while(x+y)

condition $x+y = 2+3 = 5$ is always true

So, this is infinite loop

Program 6

```
#include<iostream>
using namespace std;
```

```
int main()
```

```
{
    int i, x, y;
    x = 2;
    y = 3;
```

```
    i=2;
    while((x+y)>0)
```

```
{
    cout<<"Stephen Hawking"<<endl;
    i++;
```

```
    }
    return 0;
```

```
}
```

i=2;

while((x+y)>0)

Condition $(x+y) > 0$

$(2+3) > 0 \longrightarrow 5 > 0$ is always true.

So, this is also example of **infinite** (∞) loop

Program 7

```
#include<iostream>
using namespace std;
```

```
int main()
{
    int i, x, y;
    x = 2;
    y = 3;

    i=2;
    while((x+y)<0)
    {
        cout<<"Stephen Hawking"<<endl;
        i++;
    }

    return 0;
}
```

```
i=2;
while((x+y) < 0))
```

Condition $(x+y) > 0$

$(2+3) < 0 \longrightarrow 5 < 0$ is FALSE.

So, there is **no output**

Program 8

```
#include<iostream>
using namespace std;
```

```
int main()
{
    int i;

    i=2;
    while(i<=5)
    {
        cout<<"Stephen Hawking"<<endl;
        cout<<"Galaxy"<<endl;
        i++;
    }
    return 0;
}
```

```
i=2;
while(i<=5)
{
    cout<<"Stephen Hawking"<<endl;
    cout<<"Galaxy"<<endl;
    i++;
}
```

Value of i	Condition $i \leq 5$	Output
2	$2 \leq 5$ True	Stephen Hawking Galaxy
3	$3 \leq 5$ True	Stephen Hawking Galaxy
4	$4 \leq 5$ True	Stephen Hawking Galaxy
5	$5 \leq 5$ True	Stephen Hawking Galaxy
6	$6 \leq 5$ False	<ul style="list-style-type: none">• Loop stops here as conditions becomes false• for comes out of loop when condition becomes false• As long as condition is true, the statements inside for loop gets executed