HUE
HORUS UNIVERSITY IN EGYPT

PROGRAMMING I (CS 102)

5

LECTURE 5 LOOPS

FALL (2025-2026)

COURSE LEARNING OBJECTIVES

HUE
HORUS UNIVERSITY IN EGYPT

- Explain and define fundamental programming concepts, including variables, data types, the sequential nature of program execution, and basic program input/output (I/O).
- Trace and debug simple programs to identify and correct basic syntax and logical errors.
- **Apply basic programing concepts, such as conditional statements, loops, and functions, to solve small scale computational problems.**
- Decompose problems into smaller, manageable subroutines, and implement basic functions to enhance code modularity and reusability.
- Collaborate with peers Work in a team to design and implement a software project that meets specific functional requirements.

HUE
HORUS UNIVERSITY IN EGYPT

LECTURE 5: LOOPS

Content

1. C++ Loops

C++ LOOPS

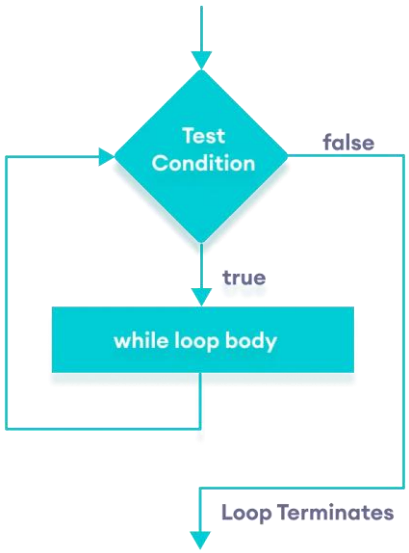
HUE
HORUS UNIVERSITY IN EGYPT

- Loops can execute a block of code as long as a specified condition is reached.
- There are **3 types** of loops in C++.
  1. **while** loop
  2. **do...while** loop
  3. **for** loop

1) C++ While Loop

The while loop loops through a block of code as long as a specified condition is true:

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



C++ LOOPS

HUE HORUS UNIVERSITY IN EGYPT

1) C++ While Loop

Example: C++ Program to print numbers from 1 to 5

```
#include <iostream>
using namespace std;
int main() {
int i = 1;
while (i < =5)
{
  cout << i << "\n";
  i++;
}
    return 0;
}
```

5

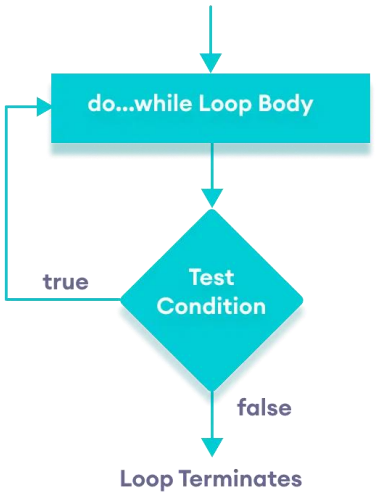
C++ LOOPS

HUE HORUS UNIVERSITY IN EGYPT

2) Do/While Loop

The do/while loop is a variant of the while loop. This loop will execute the code block once, before checking if the condition is true, then it will repeat the loop as long as the condition is true.

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



6

C++ LOOPS

HUE HORUS UNIVERSITY IN EGYPT

2) Do/While Loop Example

The example below uses a do/while loop. The loop will always be executed at least once, even if the condition is false, because the code block is executed before the condition is tested:

```
int i = 1;
do {
  cout << i << "\n";
  i++;
}
while (i <= 5);
```

7

C++ LOOPS

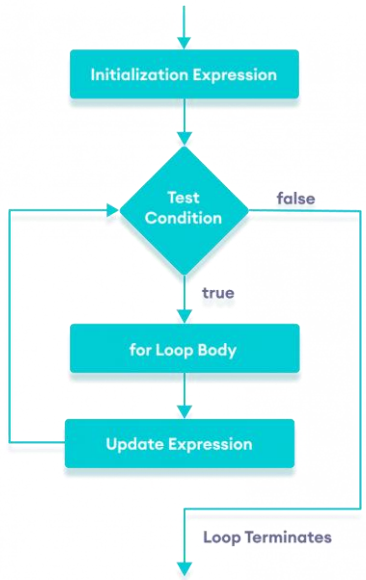
HUE HORUS UNIVERSITY IN EGYPT

3) C++ For Loop

When you know exactly how many times you want to loop through a block of code, use the for loop instead of a while loop:

```
for (initial value; condition; step)
{
    // code block to be executed
}
```

```
for (int i = 1; i <= 5; i++)
 {
  cout << i << "\n";
}
```



8

C++ LOOPS

HUE HORUS UNIVERSITY IN EGYPT

3) C++ For Loop

Example: This example will only ***print even values between 0 and 10***:

```
for (int i = 0; i <= 10; i = i + 2)
{
  cout << i << "\n";
}
```

9

C++ BREAK AND CONTINUE

HUE HORUS UNIVERSITY IN EGYPT

C++ Break

- The break statement can also be used to jump out of a loop.

C++ Continue

- The continue statement breaks one iteration (in the loop), if a specified condition occurs, and continues with the next iteration in the loop.

10

C++ BREAK AND CONTINUE

HUE HORUS UNIVERSITY IN EGYPT

C++ Break

- This example jumps out of the loop when i is equal to 4:

```
for (int i = 0; i < 10; i++) {
  if (i == 4) {
    break;
  }
  cout << i << "\n";
}
```

C++ Continue

- This example skips the value of 4:

```
for (int i = 0; i < 10; i++) {
  if (i == 4) {
    continue;
  }
  cout << i << "\n";
}
```

11

BREAK AND CONTINUE IN WHILE LOOP

HUE HORUS UNIVERSITY IN EGYPT

- You can also use break and continue in while loops:
- **Break example:**

```
int i = 0;
while (i < 10) {
  cout << i << "\n";
  i++;
  if (i == 4) {
    break;
  }
}
```

12

BREAK AND CONTINUE IN WHILE LOOP

HUE HORUS UNIVERSITY IN EGYPT

- You can also use break and continue in while loops:
- **Continue example:**

```
int i = 0;
while (i < 10) {
  if (i == 4) {
    i++;
    continue;
  }
  cout << i << "\n";
  i++;
}
```

13

EXAMPLE

HUE HORUS UNIVERSITY IN EGYPT

Example 2: Sum of Positive Numbers Only –if the user enters a negative number the loop ends

```
#include <iostream>
using namespace std;
int main() {
    int number;
    int sum = 0;
    cout << "Enter a number: ";     // take input from the user
    cin >> number;
    while (number >= 0) {
     sum += number;                 // add all positive numbers
    cout << "Enter a number: ";     // take input again if the number is positive
       cin >> number;
    }
     cout << "\nThe sum is " << sum << endl; // display the sum

       return 0;
}
```

14

EXAMPLE

HUE HORUS UNIVERSITY IN EGYPT

Example 3: Calculate the sum of numbers in a range from 10 to 1000 and not divisible by 7.

```
#include <iostream>
using namespace std;
int main() {
int sum = 0;
for (int i = 10 ; i <= 1000 ; ++i)
 {
    if (i % 7 != 0) {
       sum += i;
                  }
 }
cout << "\nThe sum is " << sum << endl; // display the sum

      return 0;
}
```

15

PRACTICE

HUE HORUS UNIVERSITY IN EGYPT

Example : Find errors.

```
int main() {
   int x = 10.5;
   cout << "x = " << x << endl;
   return 0;
}
```

```
int main() {
   int a = 5, b = 3;
   if (a = b)
      cout << "Equal";
   else
      cout << "Not Equal";
   return 0;
}
```

```
int main() {
while (i < 10) {
   cout << i *2<< "\n";
 }
   return 0;
}
```

```
int main() {
for (int i = 10; i >= 5; i++)
{
  cout << i << "\n";
}
   return 0;
}
```

16

HUE
HORUS UNIVERSITY IN EGYPT

THANK YOU!