

## BÀI THỰC HÀNH SỐ 6

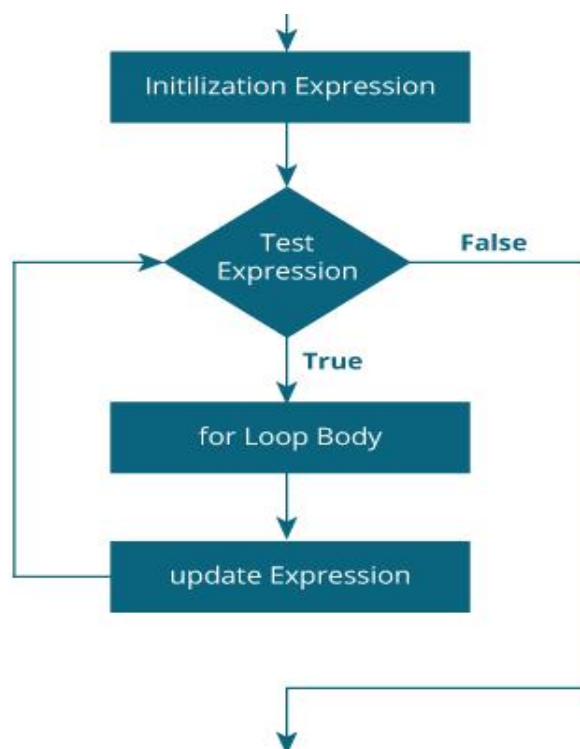
### LẶP - LOOPING

- **Loop/Iteration:** Một số lệnh được thực hiện lặp đi lặp lại
- Structure of a loop:
  - Khởi tạo - Initial block.
  - Điều kiện - Condition.
  - Các tác vụ trong mỗi lần lặp.
- Các loại lặp: lặp xác định (fixed loops), variable loops

#### I. Lặp for – for loop

The syntax of the `for` loop is:

```
1. for (initializationStatement; testExpression; updateStatement)
2. {
3.     // statements inside the body of loop
4. }
```



## Example 1: for loop

```
1.  # Print numbers from 1 to 10
2.  #include <stdio.h>
3.
4.  int main() {
5.      int i;
6.
7.      for (i = 1; i < 11; ++i)
8.      {
9.          printf("%d ", i);
10.     }
11.     return 0;
12. }
```

## Example 2: for loop

```
1.  // Program to calculate the sum of first n natural numbers
2.  // Positive integers 1,2,3...n are known as natural numbers
3.
4.  #include <stdio.h>
5.  int main()
6.  {
7.      int num, count, sum = 0;
8.
9.      printf("Enter a positive integer: ");
10.     scanf("%d", &num);
11.
12.     // for loop terminates when num is less than count
13.     for(count = 1; count <= num; ++count)
14.     {
15.         sum += count;
16.     }
17.
18.     printf("Sum = %d", sum);
19.
20.     return 0;
21. }
```

**Example 3:** Chương trình in ra bảng ASCII.

```
for (code= 0; code <256; code++)
    printf ("%c, %3d, %3o, %3X\n", code, code, code, code);
```

A,	65,	101,	41
B,	66,	102,	42
C,	67,	103,	43
D,	68,	104,	44
E,	69,	105,	45
F,	70,	106,	46
G,	71,	107,	47
H,	72,	110,	48
I,	73,	111,	49
J,	74,	112,	4A
K,	75,	113,	4B
L,	76,	114,	4C
M,	77,	115,	4D
N,	78,	116,	4E
O,	79,	117,	4F
P,	80,	120,	50
Q,	81,	121,	51
R,	82,	122,	52
S,	83,	123,	53
T,	84,	124,	54
U,	85,	125,	55

## II. Lặp while - while loop

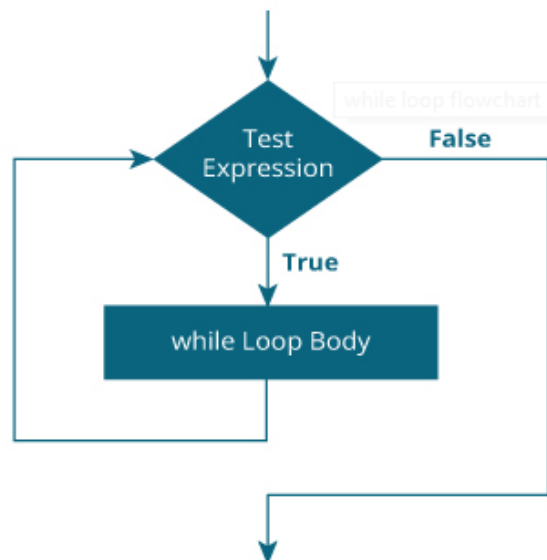
The syntax of the `while` loop is:

```

1. while (testExpression)
2. {
3.     // statements inside the body of the loop
4. }

```

### Flowchart of while loop



## Example 1: while loop

```
1. // Print numbers from 1 to 5
2.
3. #include <stdio.h>
4. int main()
5. {
6.     int i = 1;
7.
8.     while (i <= 5)
9.     {
10.        printf("%d\n", i);
11.        ++i;
12.    }
13.
14.    return 0;
15. }
```

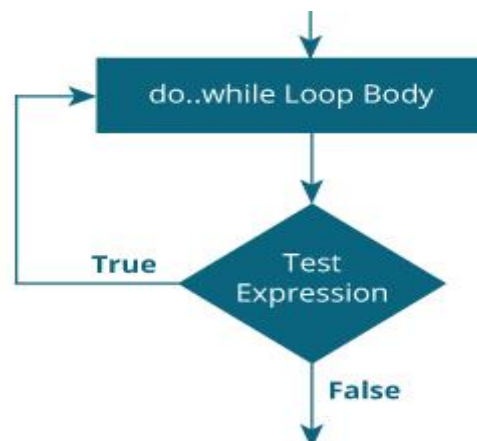
**Example 2:** Chương trình in ra bảng ASCII.

```
int code=0;
while (code<256)
{ printf ("%c, %3d, %3o, %3X\n", code, code, code, code);
  code++;
}
```

### III. Lặp do...while

The syntax of the `do...while` loop is:

```
1. do
2. {
3.     // statements inside the body of the loop
4. }
5. while (testExpression);
```



## Example 2: do...while loop

```
1. // Program to add numbers until the user enters zero
2.
3. #include <stdio.h>
4. int main()
5. {
6.     double number, sum = 0;
7.
8.     // the body of the loop is executed at least once
9.     do
10.    {
11.        printf("Enter a number: ");
12.        scanf("%lf", &number);
13.        sum += number;
14.    }
15.    while(number != 0.0);
16.
17.    printf("Sum = %.2lf",sum);
18.
19.    return 0;
20. }
```

**Example 2:** Chương trình in ra bảng ASCII.

```
do
{ printf ("%c, %3d, %3o, %3X\n", code, code, code, code);
  code++;
}
while (code <256);
```

## IV. Iteration: Break/ Bypass a loop

```
#include <stdio.h>
int main()
{ int S=0;
  int i;
  for (i=0;i<5; i++)
  {   if (i%2==1) break;
      else S+=i;
  }
  printf ("\nS=%d", S);
  for (i=0;i<5; i++)
  {   if (i%2==1) continue;
      S+=i;
  }
  printf ("\nS=%d", S);
  getchar();
  return 0;
}
```

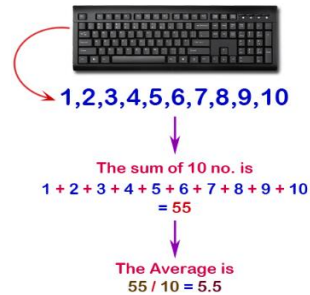
0 1 2 3 4 5

0 1 2 3 4 5

## BÀI TẬP THỰC HÀNH

### Bài 1

Viết chương trình nhập 10 số nguyên từ bàn phím. Tính tổng và trung bình của chúng.



### Bài 2

Viết chương trình tính giai thừa  $n! = 1 \times 2 \times 3 \times \dots \times n$

**Bộ dữ liệu test:**

Input	Output
0	1
1	1
2	2
3	6

### Bài 3

Viết chương trình hiển thị bảng nhân của một số nguyên nhập từ bàn phím.

Test Data :  
Input the number (Table to be calculated) : 15  
Expected Output :  
15 X 1 = 15  
...  
...  
15 X 10 = 150

### Bài 4

Viết chương trình in ra màn hình tất cả các ước số của 1 số nguyên n nhập từ bàn phím.

**Bộ dữ liệu test:**

Input	Output
8	1, 2, 4, 8
6	1, 2, 3, 6
-20	1, 2, 4, 5, 10, 20

## Bài 5

Viết chương trình tính tổng của dãy  $[9 + 99 + 999 + 9999 + \dots]$ .

Test Data :

Input the number or terms :5

Expected Output :

9 99 999 9999 99999

The sum of the series = 111105



## Bài 6

Viết chương trình để xác định xem một số đã cho có phải là số nguyên tố hay không.

Test Data :

Input a number: 13

Expected Output :

13 is a prime number.

## Bài 7

Viết chương trình tìm ước số chung lớn nhất của 2 số nguyên dương.

**Bộ dữ liệu test:**

Input		Output
Số thứ 1	Số thứ 2	Ước số chung lớn nhất
8	12	4
6	12	6
7	1	1

## Bài 8

Viết chương trình tính tổng các số nguyên được nhập từ bàn phím cho đến khi nhập 0 thì dừng.

### **Suggested algorithm**

Begin

    S=0;

    do

    {

        Nhập x;

        If (x != 0)

            S = S + x;

    }

    while (x!=0);

    Xuất S;

End

## Bài 9

### **Problem**

- Viết chương trình cho người dùng nhập vào các ký tự cho đến khi nhấn phím ENTER (mã 10). Chương trình sẽ in ra số lượng chữ số, chữ cái, và số lượng các phím khác đã nhập.

#define ENTER 10

### **Suggested algorithm**

Begin

    int noDigits = 0, noLetters = 0, noOthers = 0;

    char c=0;

    printf("Enter a string:");

    while (c!= ENTER)

    {

        Nhập c;

        if ( c>='0' && c <='9')

            noDigits++;

        else if ( (c>='a' && c <='z') || (c>='A' && c <='Z') )

            noLetters++;

        else

            noOthers++;

    }

    Xuất: noDigits, noLetters, noOthers

End



## Bài 10

<b>Problem</b>	<ul style="list-style-type: none"><li>- Viết chương trình cho người dùng nhập vào các ký tự cho đến khi nhấn phím ENTER (mã 10). Chương trình sẽ in ra số lượng nguyên âm, phụ âm, và số lượng các phím khác đã nhập.</li></ul>
<b>Suggested algorithm</b> Begin  do { Nhập ch; Chuyển ch sang in hoa if (ch>='A' && ch <='Z') { switch (ch) { case 'A' : case 'E' : case 'I' : case 'O' : case 'U' : nVowels ++; break; default: nConsonants++; } } else if (ch!=10) nOthers++; } while ( ch != '\n'); Xuất nVowels, nConsonants, nOthers; End	