

NỘI DUNG

- Khái niệm về vòng lặp (cấu trúc lặp - loop)
- Sử dụng while, do while
- For và vòng lặp lồng nhau
- Ôn tập Hướng dẫn viết console app với câu lệnh điều kiện

```
C# Program.cs X
app > C# Program.cs

1 int counter = 0; // Khởi tạo biến đếm với giá trị ban đầu là 0
2 // Vòng lặp while sẽ tiếp tục lặp khi điều kiện (counter < 5) còn đúng
3 while (counter < 5)
4 {
5     Console.WriteLine("Counter value: " + counter); // In giá trị biến đếm ra màn hình
6     counter++; // Tăng giá trị của counter lên 1 sau mỗi lần lặp
7 }
8
9 Console.WriteLine("Vòng lặp đã kết thúc!"); // Thông báo khi vòng lặp đã kết thúc
```



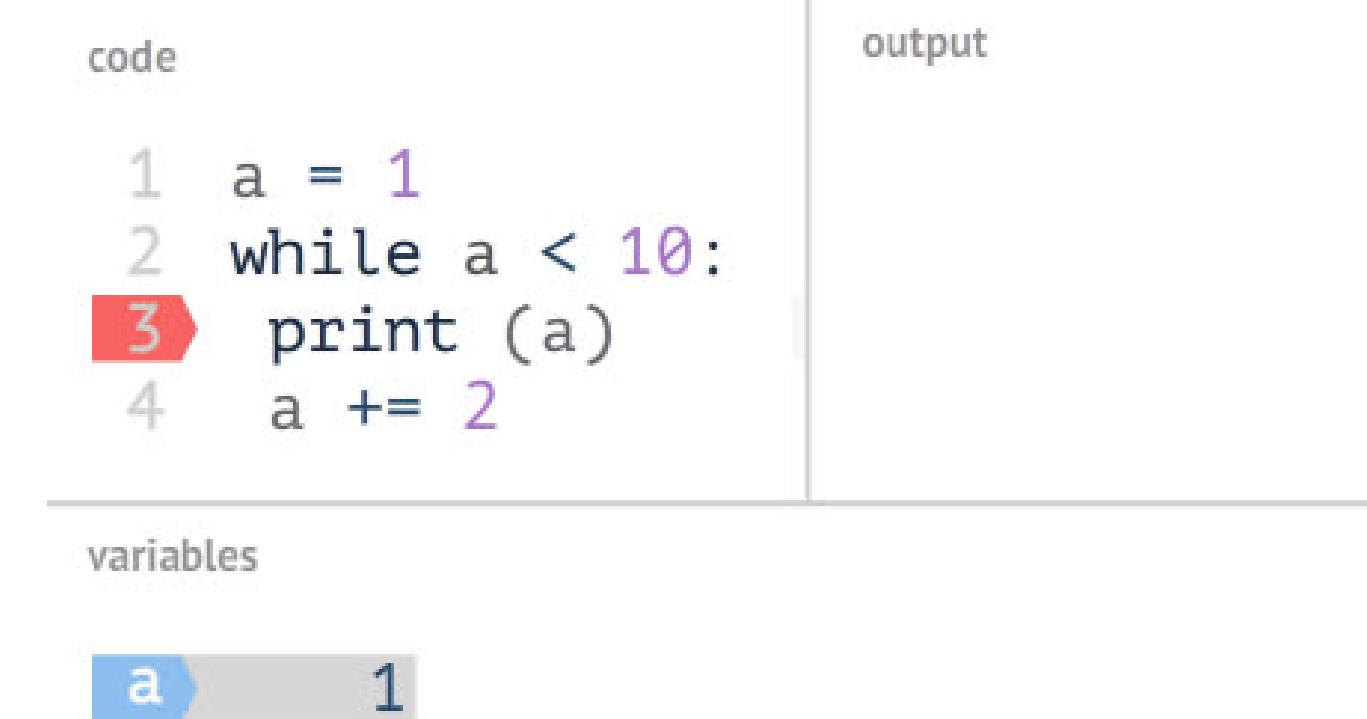


KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

CẤU TRÚC LẶP

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

- Trong lập trình đôi lúc ta cần xử lý các công việc lặp đi lặp lại nhiều lần.
- Lấy 1 ví dụ:
 - + Đếm ngược thời gian trong game xét hết giờ
 - + Vàng tăng lên mỗi giây
 - + Thời gian hồi chiêu thức trong game hay thời gian biến về ...



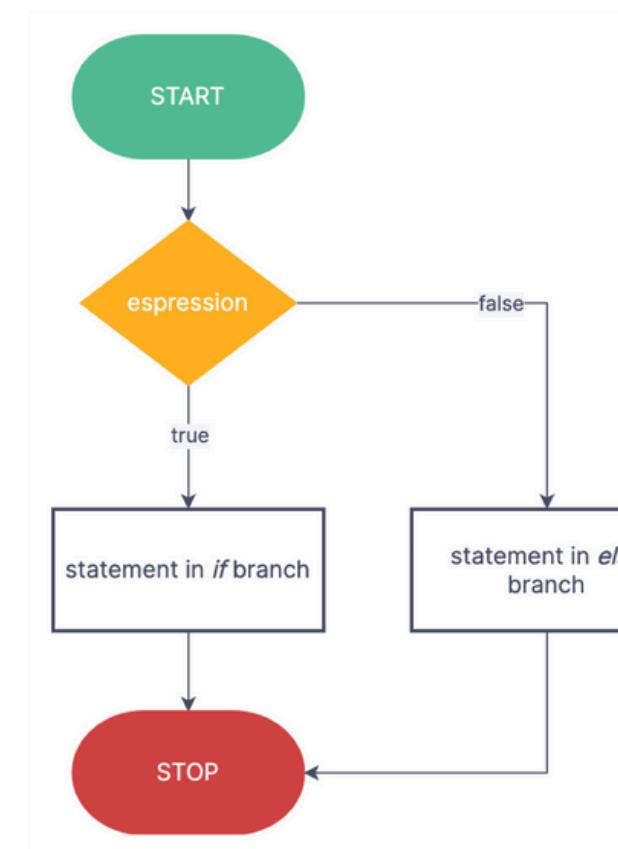


KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

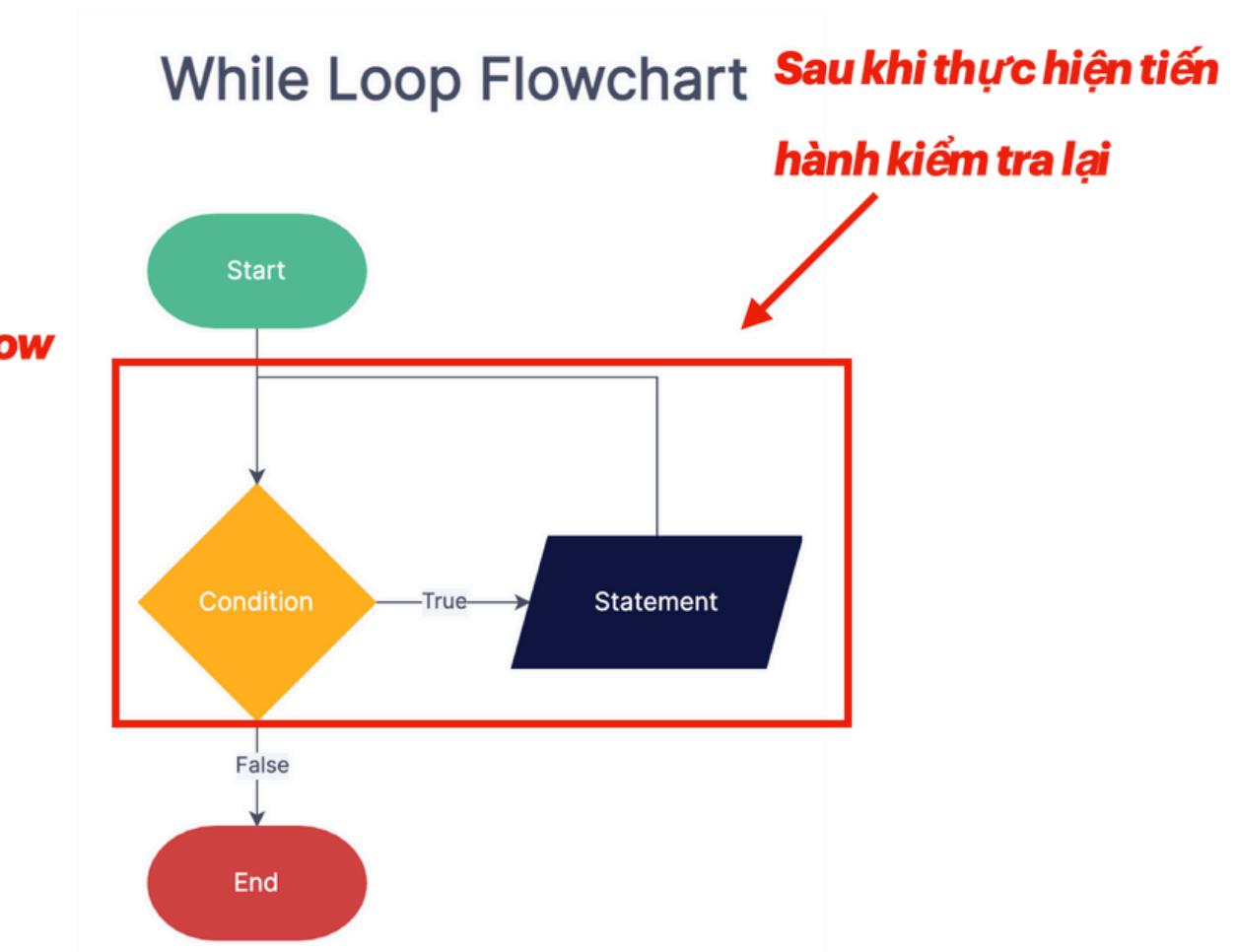
CẤU TRÚC LẶP (FOR , WHILE)

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

If flow



while flow





KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

CẤU TRÚC LẶP (FOR , WHILE)

Lấy ví dụ trong lập trình để in ra màn hình 6 lần với cùng 1 nội dung ta sẽ làm gì ?

-> Copy paste đoạn code 6 lần ? vậy 1.000 lần 1.000.000 lần thì sao ?

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

The screenshot shows a computer monitor displaying a C# code editor. The title bar of the window says "1. Khái niệm cơ bản trong lập trình". Below the title bar, there is a section titled "◆ Một số ví dụ" (Some examples). Inside this section, there are two lines of C# code:

```
Console.WriteLine("Hello Dev Khải");
Console.WriteLine("Hello Dev Khải");
```

To the right of the code editor, there is a "Output" window. The output window displays the following text:
Hello techx
Hello tec|



SỬ DỤNG WHILE, DO WHILE

CẤU TRÚC LẶP (FOR , WHILE)

Ví dụ: In ra 1 số từ 0 đến 4 với vòng lặp while

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

The illustration shows a white computer monitor with a slide displayed on its screen. The slide has a yellow header 'Khái niệm cơ bản trong lập trình' and a blue sub-header '♦ Cấu trúc lặp (for , while)'. Below this, there is a note: '- Đối với vòng lặp while khi biểu diễn khó hơn for 1 tí. Bắt buộc phải xác định 4 yếu tố.' A code snippet is shown in a teal box:

```
int count = 0;  
while (count < 5 ) {  
    Console.WriteLine(x);  
    count += 1;  
}
```

To the right of the monitor, there is a vertical 'Output' window which is currently empty.



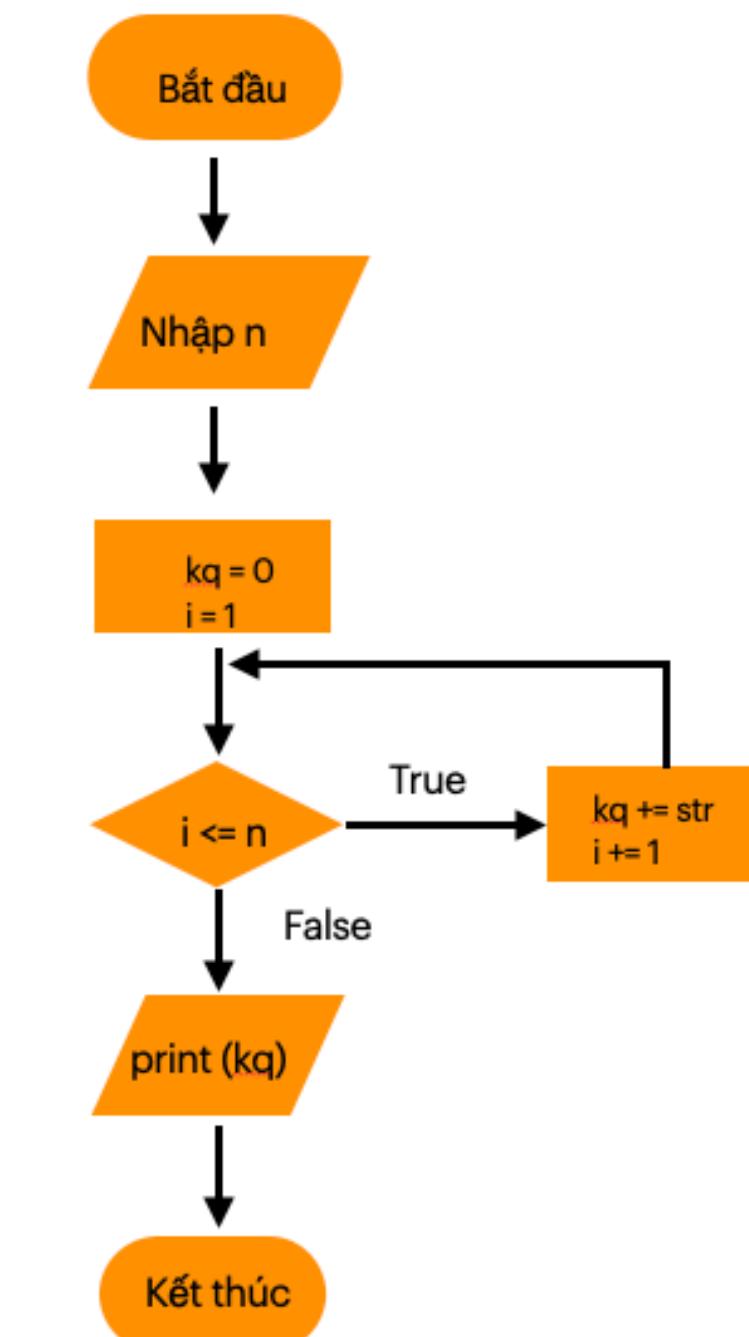
KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

CẤU TRÚC LẶP (FOR , WHILE)

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

◆ Một số ví dụ

- Viết chương nhập cho phép người dùng nhập vào 1 chuỗi và số lần xuất hiện. In chuỗi đó ra màn hình tương ứng





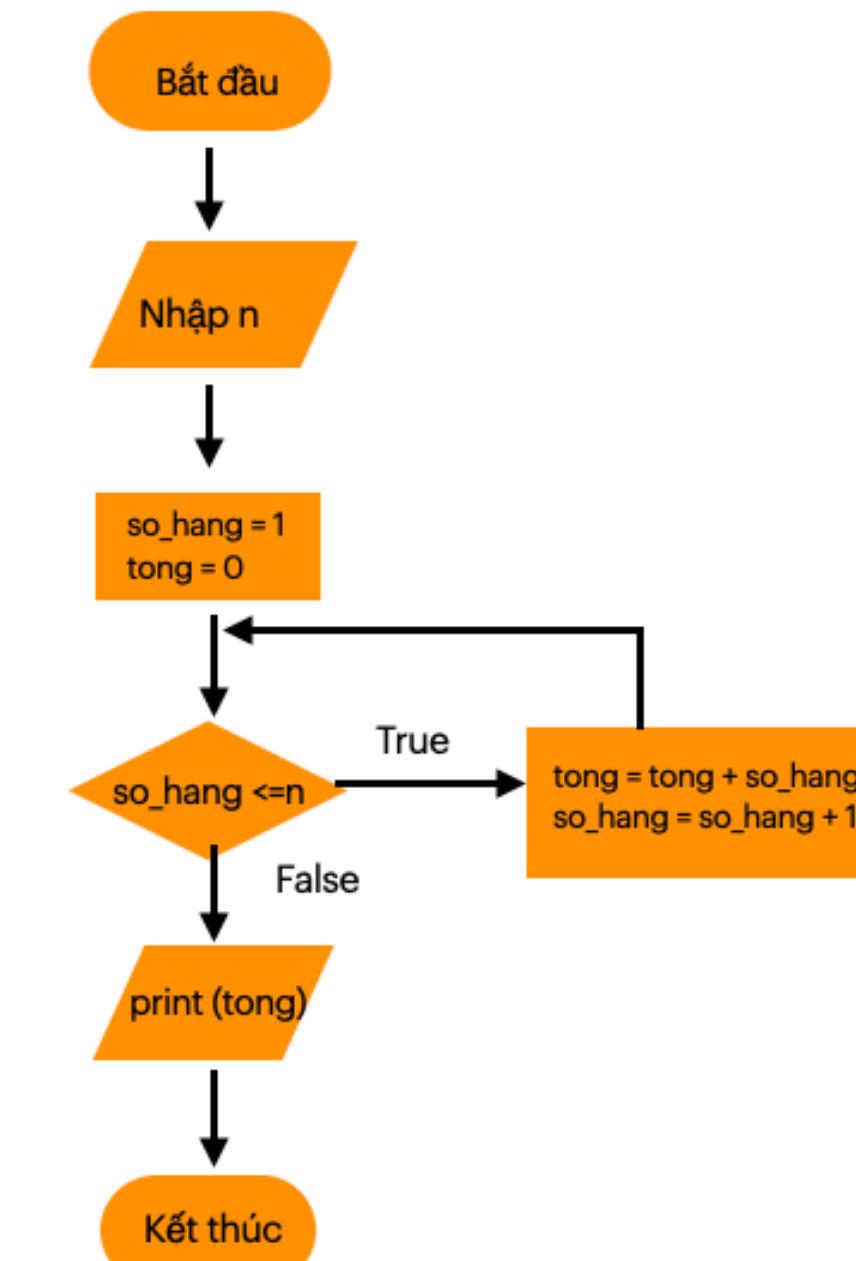
KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

CẤU TRÚC LẶP (FOR , WHILE)

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

❖ Một số ví dụ

“ “
1. $S = 1 + 2 + 3 + \dots + (n-2) + (n-1) + n$





KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

BÀI TẬP

1

2

3

4

5

6

7

8

9

10

BÀI TẬP VÒNG LẶP 1

“
Viết chương trình cho phép người dùng
nhập vào 1 số n, tính tổng các số từ 1 -> n

$$S = 1 + 2 + 3 + \dots + (n-2) + (n-1) + n$$

- **Nhập (Input):**
 - Input của bài toán này là số "n" (đây là số nguyên dương), tức là số cuối cùng trong dãy số liên tiếp bắt đầu từ 1.
- **Xuất (Output):**
 - Output là tổng "S" của dãy số từ 1 đến "n".
 - Ví dụ:
 - Input:
 - n = 5
 - Output:
 - S = 15
 - Giải thích:
 - Trong ví dụ này, số "n" là 5 (đây là input).
 - Chúng ta cần tính tổng các số từ 1 đến 5, tức là: S = 1 + 2 + 3 + 4 + 5.
 - chúng ta nhận được kết quả là 15 (đây là output).



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

BÀI TẬP

1

2

3

4

5

6

7

8

9

10

BÀI TẬP VÒNG LẶP 2

“
Viết chương trình cho phép người dùng
nhập vào 1 số n, tính tổng các số từ 2 -> 2n

$$S = 2 + 4 + 6 + \dots + 2n$$

- **Nhập (Input):**
 - Input của bài toán này là số "n" (đây là số nguyên dương), tức là số cuối cùng trong dãy số liên tiếp bắt đầu từ 2.
- **Xuất (Output):**
 - Output là tổng "S" của dãy số từ 2 đến "2n".
- **Ví dụ:**
 - **Input:**
 - n = 5
 - **Output:**
 - S = 30
- **Giải thích:**
 - Trong ví dụ này, số "n" là 5 (đây là input).
 - Chúng ta cần tính tổng các số từ 1 đến $5 * 2$, tức là: $S = 2 + 4 + 6 + 8 + 10$.
 - Chúng ta nhận được kết quả là 30 (đây là output).



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

BÀI TẬP

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10

BÀI TẬP VÒNG LẶP 3

“

Viết chương trình cho phép người dùng nhập vào số n, tính n! (n giai thừa)

$$n! = 1 * 2 * 3 * 4 * \dots * (n-1) * n$$

- **Mô tả:**
 - Giai thừa của 1 số được tính từ tích các số từ 1 đến số đó, trường hợp đặc biệt $0! = 1$
- **Nhập (Input):**
 - Input của bài toán này là số "n" (đây là số nguyên dương).
- **Xuất (Output):**
 - Output là giai thừa số đó được tính bằng công thức: $n! = 1 * 2 * 3 * 4 * \dots * (n-1) * n$. Trường hợp đặc biệt $0! = 1$
- **Ví dụ:**
 - **Input:**
 - $n = 5$
 - **Output:**
 - $n! = 1 * 2 * 3 * 4 * 5$ hoặc $n! = 5 * 4 * 3 * 2 * 1$
- **Giải thích:**
 - Trong ví dụ này, số "n" là 5 (đây là input).
 - Chúng ta cần tính tích các số từ 1 đến 5 tức là: $5! = 1 * 2 * 3 * 4 * 5$.
 - Chúng ta nhận được kết quả là 120 (đây là output). Trường hợp đặc biệt $0! = 1$



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

BÀI TẬP

1

2

3

4

5

6

7

8

9

10

BÀI TẬP VÒNG LẶP 4

“

Viết chương trình tính tổng bình phương các số từ 1 đến n.

$$S = 1^2 + 2^2 + \dots + n^2$$

- **Mô tả:**
 - Để tính bình phương của 1 số ví dụ : 1^2 ta có thể dùng $1^{**} 2$
- **Nhập (Input):**
 - Input của bài toán này là số "n" (đây là số nguyên dương).
- **Xuất (Output):**
 - Output là tổng bình phương của các số từ 1 đến n: $S = 1^2 + 2^2 + \dots + n^2$.
- **Ví dụ:**
 - **Input:**
 - $n = 3$
 - **Output:**
 - $S = 1^2 + 2^2 + 3^2 = 14$
- **Giải thích:**
 - Trong ví dụ này, số "n" là 3 (đây là input).
 - Chúng ta cần tính tổng các số từ 1 đến n^2 : $S(3) = 1^2 + 2^2 + 3^2 = 14$
 - Chúng ta nhận được kết quả là 14 (đây là output).



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

BÀI TẬP

1

2

3

4

5

6

7

8

9

10

BÀI TẬP VÒNG LẶP 5

BÀI TẬP

“
Viết chương trình nhập vào số n cho biết
số n có phải là số nguyên tố hay không ?
n là số nguyên tố

- **Mô tả:**
 - Số nguyên tố là số có 2 ước số.
- **Ước số là gì?**
 - Ước số là các số mà số đó chia hết.
- **Lấy ví dụ:**
 - + 7 là số nguyên tố, 7 chia hết cho 1 và 7 (1 và 7 là ước số của 7).
 - + 10 không phải là số nguyên tố, 10 chia hết cho 1,2,5,10. 10 có 4 ước số nên 10 không phải là số nguyên tố
- **Nhập (Input):**
 - Input của bài toán này là số "n" (đây là số nguyên dương).
- **Xuất (Output):**
 - n là số nguyên tố hoặc n không phải là số nguyên tố
- **Ví dụ:**
 - **Input:**
 - n = 7 hoặc n = 10
 - **Output:**
 - 7 là số nguyên tố hoặc 10 không phải là số nguyên tố



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

12

13

14

15

16

17

18

19

20

FOR VÀ VÒNG LẶP LÔNG NHAU

```
for (khởi_tạo; điều_kiện; thay_đổi_biên_đếm)
{
    // Khối lệnh được thực thi trong mỗi lần lặp
}
```

```
for (khởi_tạo; điều_kiện; thay_đổi_biên_đếm)
{
    // Khối lệnh được thực thi trong mỗi lần lặp
}

Khối lệnh thực thi trong vòng lặp có thể là vòng lặp
```

SỰ TƯƠNG QUAN VỀ FOR VÀ WHILE

Có thể xem for như 1 cách viết ngắn gọn của while vì các thành phần tương đương.

```
C# Program.cs ×
app > C# Program.cs
1 int i = 0; // 1. Khởi tạo biến
2
3 while (i <= 5) // 2. Điều kiện
4 {
5     Console.WriteLine("Đếm lên: " + i); // 4. Khối lệnh
6     i++; // 3. Thay đổi biến đếm
7 }
```

```
C# Program.cs ×
app > C# Program.cs
1 // Đếm từ 0 đến 5
2 for (int i = 0; i <= 5; i++) // 1. Khởi tạo biến, 2. Điều kiện, 3. Thay đổi biến đếm
3 {
4     Console.WriteLine("Đếm lên: " + i); // 4. Khối lệnh
5 }
6
7
8
9
```



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

12

13

14

15

16

17

18

19

20

DUYỆT FOR MỞ RỘNG

- For là cú pháp dùng để duyệt tuần tự biến thay đổi dựa trên điều kiện vì vậy để thay đổi 1 giá trị có nhiều cách tuy nhiên 2 cách thay đổi giá trị phổ biến đó là tăng và giảm. Người ta thường gọi là duyệt xuôi hoặc duyệt ngược.

VÍ DỤ:

- Dưới đây là 1 ví dụ về cách giải quyết 1 bài toán giai thừa theo hai hướng là duyệt các số hạng tăng dần từ 1 -> 5 , cách còn lại là duyệt từ 5 về 1.
- Đối với 1 số bài toán đặt biệt trong 1 lần xử lý có thể vừa thực hiện duyệt xuôi và ngược cùng lúc để tối ưu hiệu suất**

```
// Tính giai thừa từ 1 đến 5
Console.WriteLine("Giai thừa từ 1 đến 5:");
for (int i = 1; i <= 5; i++)
{
    int factorial = 1;
    for (int j = 1; j <= i; j++)
    {
        factorial *= j;
    }
    Console.WriteLine($"{i}! = {factorial}");
}
```

DUYỆT XUÔI

```
// Tính giai thừa từ 5 về 1
Console.WriteLine("Giai thừa từ 5 về 1:");
for (int i = 5; i >= 1; i--)
{
    int factorial = 1;
    for (int j = 1; j <= i; j++)
    {
        factorial *= j;
    }
    Console.WriteLine($"{i}! = {factorial}");
}
```

DUYỆT NGƯỢC



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

12

13

14

15

16

17

18

19

20

VÍ DỤ VỀ VÒNG LẶP LỒNG NHAU

BÀI TẬP VÒNG LẶP 6

Viết chương trình cho phép nhập vào số dòng và số cột. In ra ma trận sao tương ứng

✓ $r = 3, c = 5$



Mô tả:

Bài tập yêu cầu người dùng nhập vào số nguyên dòng và số nguyên cột viết chương trình in ra các hàng và cột chứa các ngôi sao tương ứng.

Nhập (Input):

so_dong = 3, so_cot = 5

Xuất (Output):





KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

12

13

14

15

16

17

18

19

20

VÍ DỤ VỀ VÒNG LẶP LỒNG NHAU

BÀI TẬP VÒNG LẶP 7

“
Viết chương trình nhập vào số nguyên n. In ra tam giác vuông tương ứng. Ví dụ: n = 3.
 *
 * *
 * * *
 ”

Mô tả:

Bài tập yêu cầu người dùng nhập vào 1 con số nguyên (n), ví dụ người dùng nhập vào n=3 thì in ra tam giác vuông như hình trên

Nhập (Input):

n=3

Xuất (Output):

 *
 * *
 * * *



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

12

13

14

15

16

17

18

19

20

VÍ DỤ VỀ VÒNG LẶP LỒNG NHAU

BÀI TẬP VÒNG LẶP 8

Mô tả:

Bài tập yêu cầu người dùng nhập vào 1 con số nguyên (n), ví dụ người dùng nhập vào n=3 thì in ra tam giác cân như hình trên

Nhập (Input):

n=3

Xuất (Output):

“
Viết chương trình nhập vào số nguyên n
xuất ra tam giác cân tương ứng






KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

12

13

14

15

16

17

18

19

20

VÍ DỤ VỀ VÒNG LẶP LỒNG NHAU

BÀI TẬP VÒNG LẶP 8

Mô tả:

Bài tập yêu cầu người dùng nhập vào 1 con số nguyên (n), ví dụ người dùng nhập vào n=3 thì in ra tam giác cân như hình trên

Nhập (Input):

n=3

Xuất (Output):

“
Viết chương trình nhập vào số nguyên n
xuất ra tam giác cân tương ứng






KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

12

13

14

15

16

17

18

19

20

BÀI TẬP TỔNG HỢP

Bài 1: In bảng cửu chương

- Viết chương trình sử dụng vòng lặp for để in bảng cửu chương của một số bất kỳ (ví dụ: 5).
- Gợi ý:
 - Yêu cầu người dùng nhập vào một số nguyên n.
 - Sử dụng vòng lặp for để in ra bảng cửu chương của số đó từ 1 đến 10.

Bài 2: Tìm số nguyên tố trong khoảng từ 1 đến đến số người dùng nhập vào

- Viết chương trình sử dụng vòng lặp for để tìm tất cả các số nguyên tố trong khoảng từ 1 đến 50.
- Gợi ý:
 - Sử dụng một vòng lặp for để lặp qua các số từ 2 đến n.
 - Trong mỗi lần lặp, sử dụng một vòng lặp for khác để kiểm tra xem số đó có phải là số nguyên tố không.
 - In ra các số nguyên tố tìm được.

Bài 3: Đảo ngược chuỗi

- Viết chương trình sử dụng vòng lặp for để đảo ngược một chuỗi.
- Gợi ý:
 - Yêu cầu người dùng nhập vào một chuỗi.
 - Sử dụng vòng lặp for để lặp qua từng ký tự của chuỗi từ cuối đến đầu.
 - Tạo một chuỗi mới chứa các ký tự đảo ngược và in ra kết quả.



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

12

13

14

15

16

17

18

19

20

MỞ RỘNG ỨNG DỤNG VÒNG LẶP XỬ LÝ CHUỖI

- Một số hàm xử lý chuỗi

`str = "Hello World" => [H,e,l,l,o,W,o,r,l,d];`

.Length = 6;
str[0] = H;
str[1] = E;
...
str[Length-1] = d

C#	Kết quả	Ý nghĩa
<code>str.Contains("World")</code>	True	Kiểm tra xem chuỗi có chứa chuỗi con hay không
<code>str.StartsWith("Hello")</code>	True	Kiểm tra xem chuỗi có bắt đầu bằng chuỗi con không
<code>str.EndsWith("World")</code>	True	Kiểm tra xem chuỗi có kết thúc bằng chuỗi con không
<code>str.IndexOf('o')</code>	4	Tìm vị trí xuất hiện đầu tiên của ký tự 'o' trong chuỗi
<code>str.LastIndexOf('o')</code>	7	Tìm vị trí xuất hiện cuối cùng của ký tự 'o' trong chuỗi
<code>str.Substring(0, 5)</code>	"Hello"	Lấy chuỗi con từ vị trí chỉ định với độ dài xác định
<code>string.Join(" ", words)</code>	"Hello World"	Kết hợp các phần tử của mảng thành chuỗi với ký tự phân cách
<code>str.Remove(5)</code>	"Hello"	Xóa các ký tự từ vị trí chỉ định
<code>str.Insert(5, " Beautiful")</code>	"Hello, Beautiful World"	Chèn chuỗi vào vị trí chỉ định
<code>Regex.Replace(str, @"\d", "")</code>	"abc"	Thay thế các chuỗi phù hợp với biểu thức chính quy
<code>string.IsNullOrEmpty(str1)</code>	True	Kiểm tra chuỗi có rỗng hoặc null
<code>string.IsNullOrWhiteSpace(str2)</code>	True	Kiểm tra chuỗi có rỗng hoặc chỉ chứa khoảng trắng
<code>string.Format("Name: {0}, Age: {1}", name, age)</code>	"Name: John, Age: 30"	Định dạng chuỗi với các tham số
<code>string.Compare(str1, str2)</code>	-1	So sánh hai chuỗi, trả về giá trị khác biệt
<code>str.PadLeft(10)</code>	" Hello"	Thêm ký tự vào bên trái chuỗi để đạt độ dài cụ thể
<code>str.PadRight(10)</code>	"Hello "	Thêm ký tự vào bên phải chuỗi để đạt độ dài cụ thể
<code>str.ToCharArray()</code>	{'H', 'e', 'l', 'l', 'o'}	Chuyển đổi chuỗi thành mảng các ký tự



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

BÀI TẬP VỀ CHUỖI

12

BÀI TẬP 1: IN CHUỖI ĐẢO NGƯỢC

13

Yêu cầu

14

- Viết chương trình cho phép người dùng nhập vào 1 chuỗi ký tự bất kỳ. In ra chuỗi đảo ngược tương ứng.

15

Ví dụ :
Nhập vào: hello techx
in ra: xhcet olleh

16

17

- Lập sơ đồ 3 khối xác định input output process
- Viết code giải quyết bài toán không dùng thư viện
hay các hàm xử lý chuỗi có sẵn ví dụ: `s[::-1]`
- Hãy dùng vòng lặp để tư duy xử lý

18

19

20



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

12

13

14

15

16

17

18

19

20

BÀI TẬP VỀ CHUỖI

BÀI TẬP 2: KIỂM TRA CHUỖI ĐỐI XỨNG

Yêu cầu

- Viết chương trình cho phép người dùng nhập vào 1 chuỗi ký tự bất kì. In ra kết quả chuỗi đó có phải là chuỗi đối xứng không ?

Ví dụ :

Nhập vào: tenet
in ra: Đây là chuỗi đối xứng

Ví dụ :

Nhập vào: techx
in ra: Đây không phải là chuỗi đối xứng

- Lập sơ đồ 3 khối xác định input output process
- Viết code giải quyết bài toán không dùng thư viện hay các hàm xử lý chuỗi có sẵn ví dụ: `s[::-1]`
- Hãy dùng vòng lặp để tư duy xử lý



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

BÀI TẬP VỀ CHUỖI

12

BÀI TẬP 3: ĐẾM SỐ LƯỢNG TỪ TRONG CÂU

 **Yêu cầu**

- Viết chương trình cho phép người dùng nhập vào 1 câu đếm số lượng từ trong câu đó.

13

14

15

16

17

18

19

20

Ví dụ :
Nhập vào:toi la lap trinh vien
in ra:5

- Lập sơ đồ 3 khối xác định input output process
- Viết code giải quyết bài toán không dùng thư viện hay các hàm xử lý chuỗi có sẵn
- Hãy dùng vòng lặp để tư duy xử lý



KHÁI NIỆM VỀ VÒNG LẶP (CẤU TRÚC LẶP - LOOP)

11

12

13

14

15

16

17

18

19

20

BÀI TẬP VỀ CHUỖI

BÀI TẬP 4: LOẠI BỎ KHOẢNG TRẮNG CỦA CHUỖI

Yêu cầu

- Viết chương trình cho phép người dùng nhập vào 1 chuỗi, in ra chuỗi đó được loại bỏ khoảng trắng

Ví dụ :

Nhập vào: chuoi nay khong co khoang trang
in ra: chuoinaykhongcokhoangtrang

- Lập sơ đồ 3 khối xác định input output process
- Viết code giải quyết bài toán không dùng thư viện hay các hàm xử lý chuỗi có sẵn ví dụ: str.replace(" ","")
- Hãy dùng vòng lặp để tự duy xử lý