

# CHƯƠNG 3

## CẤU TRÚC ĐIỀU KHIỂN

2

## Cấu trúc điều khiển

### TUẦN TỰ

Lệnh 1;  
Lệnh 2;  
Lệnh 3;  
....

### RỄ NHÁNH CÓ ĐIỀU KIỆN

if  
if ... else

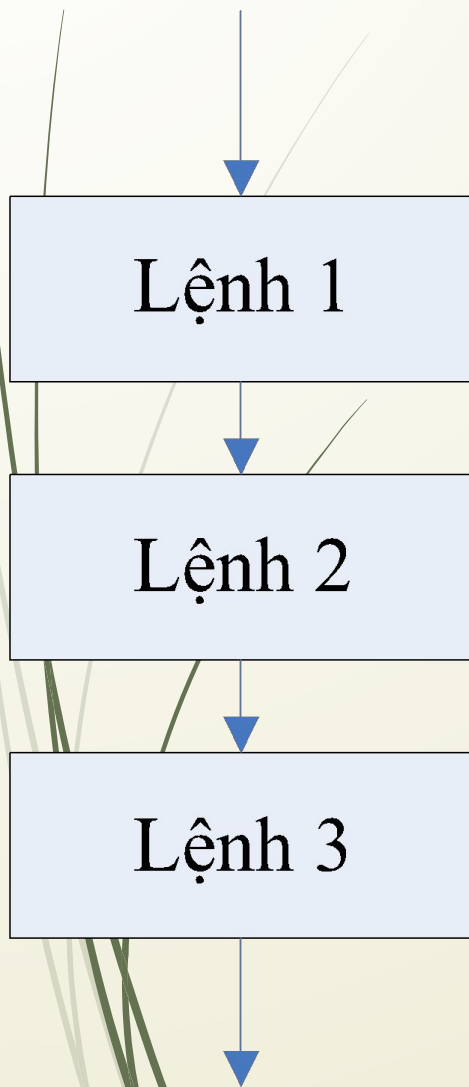
### LỰA CHỌN

switch ... case

### LẶP

for  
while  
do ... while  
foreach

## Cấu trúc tuần tự



Tuần tự thực thi tiến trình, mỗi lệnh được thực thi theo một chuỗi từ trên xuống

Xong lệnh này rồi chuyển xuống lệnh kế tiếp

```
static void Main(string []args)
```

```
{
```

```
    int a, b, tong, hieu, tich;
```

```
    float thuong;
```

```
    Console.Write ("Nhap vao a: ");
```

```
    a = int.Parse (Console.ReadLine());
```

```
    Console.Write ("Nhap vao b: ");
```

```
    b = int.Parse (Console.ReadLine());
```

```
    tong = a + b;
```

```
    hieu = a - b;
```

```
    tich = a * b;
```

```
    thuong = (float)a / b; //Ép kiểu
```

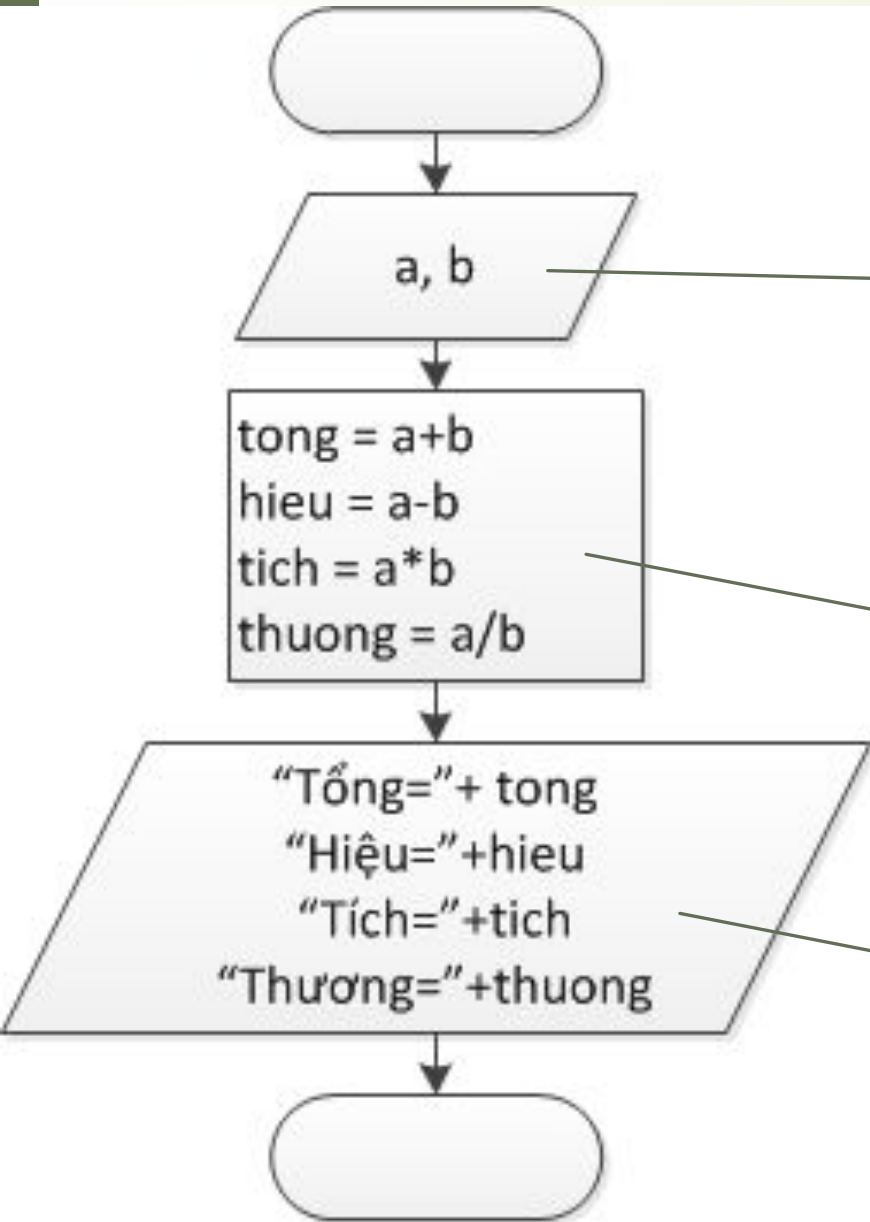
```
    Console.WriteLine ("Tong: " + tong);
```

```
    Console.WriteLine ("Hieu:" + hieu);
```

```
    Console.Write ("Tich: " + tich);
```

```
    Console.Write ("Thuong: " + thuong);
```

```
}
```



## Cấu trúc rẽ nhánh

Cấu trúc rẽ nhánh chỉ cho phép thực hiện một dãy lệnh nào đó dựa vào kết quả của một điều kiện (biểu thức quan hệ hay biểu thức so sánh)

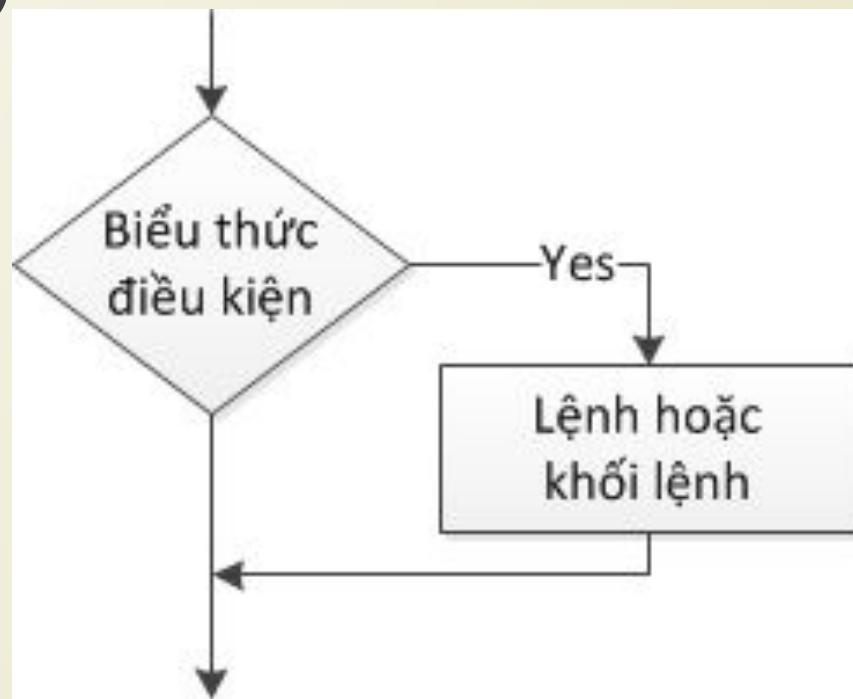
Chỉ xét trường hợp đúng

*if (biểu thức điều kiện)*

{

*<khối lệnh> ;*

}



*Nếu biểu thức điều kiện cho kết quả **true** thì thực hiện khối lệnh bên trong **if**.*

Ví dụ: Viết chương trình nhập vào một số nguyên n, in ra giá trị tuyệt đối của n

6

```
static void Main(string []args)
```

```
{
```

```
    int n;
```

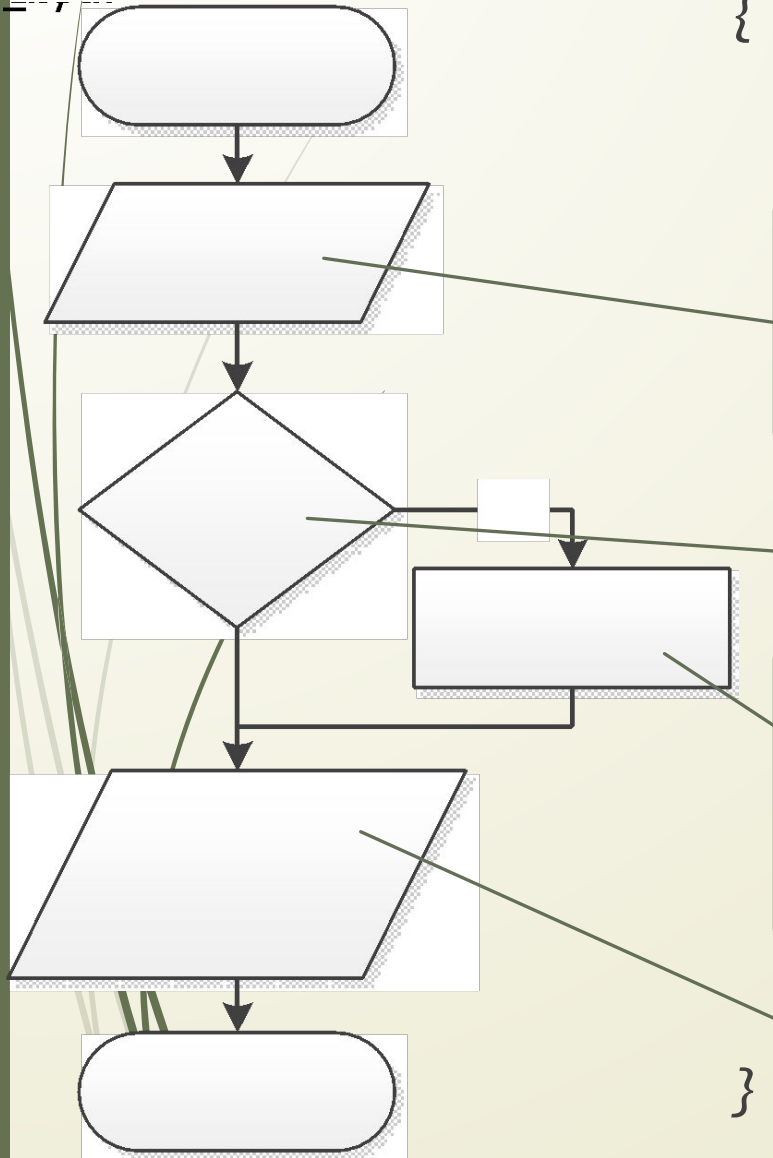
```
    Console.Write ("Nhap mot so nguyen: ");  
    n = int.Parse (Console.ReadLine());
```

```
    if (n<0)
```

```
    {  
        n=-n;  
    }
```

```
    Console.WriteLine ("Gia tri tuyet doi: "+n);
```

```
}
```



Xét cả hai trường hợp đúng và sai:

7

*if (biểu thức điều kiện)*

{

*<khối lệnh 1>;*

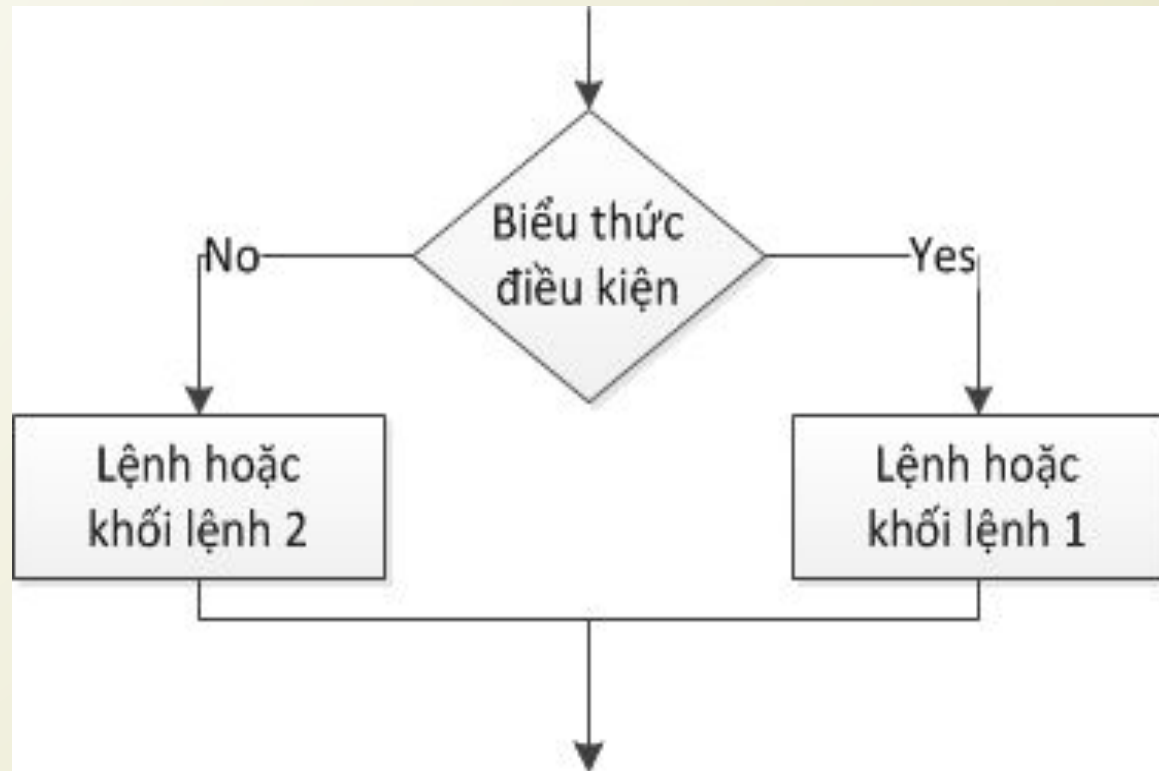
}

*else*

{

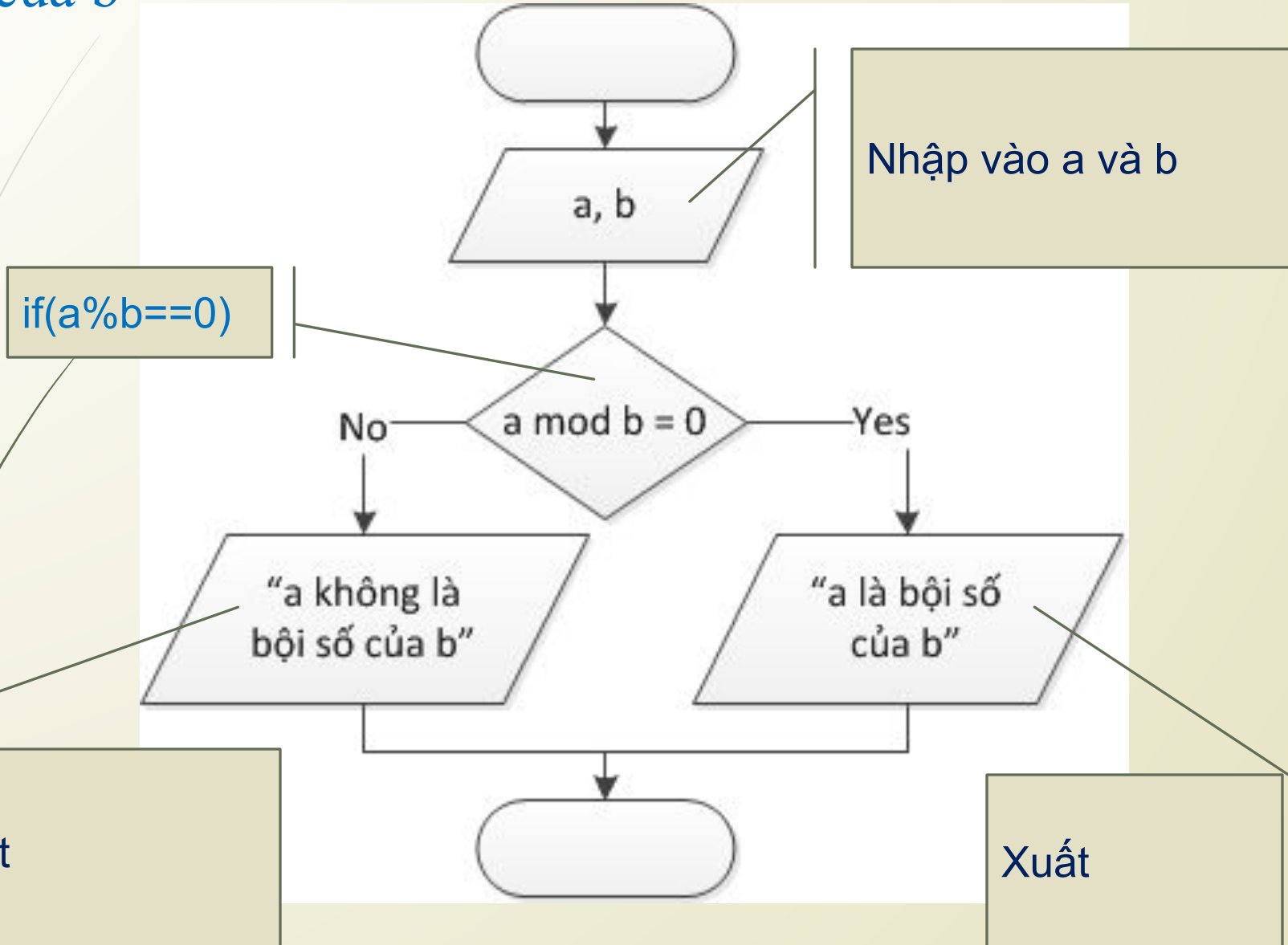
*<khối lệnh 2>;*

}



*Nếu biểu thức điều kiện cho kết quả **true** thì thực hiện khối lệnh 1, ngược lại thì cho thực hiện khối lệnh thứ 2*

VD: Nhập vào 2 số nguyên a và b, nếu a là bội số của b thì in thông báo “a là bội số của b”, ngược lại in “a không là bội số của b”





```
static void Main(string []args)
```

9

```
{
```

```
    int a, b;
```

```
    Console.Write (“Nhap vao a: “);
```

```
    a = int.Parse (Console.ReadLine());
```

```
    Console.Write(“Nhap vao b: “);
```

```
    b = int.Parse(Console.ReadLine());
```

```
    if(a%b == 0)
```

```
    {
```

```
        Console.WriteLine (“a la boi so cua b”);
```

```
    }
```

```
    else
```

```
    {
```

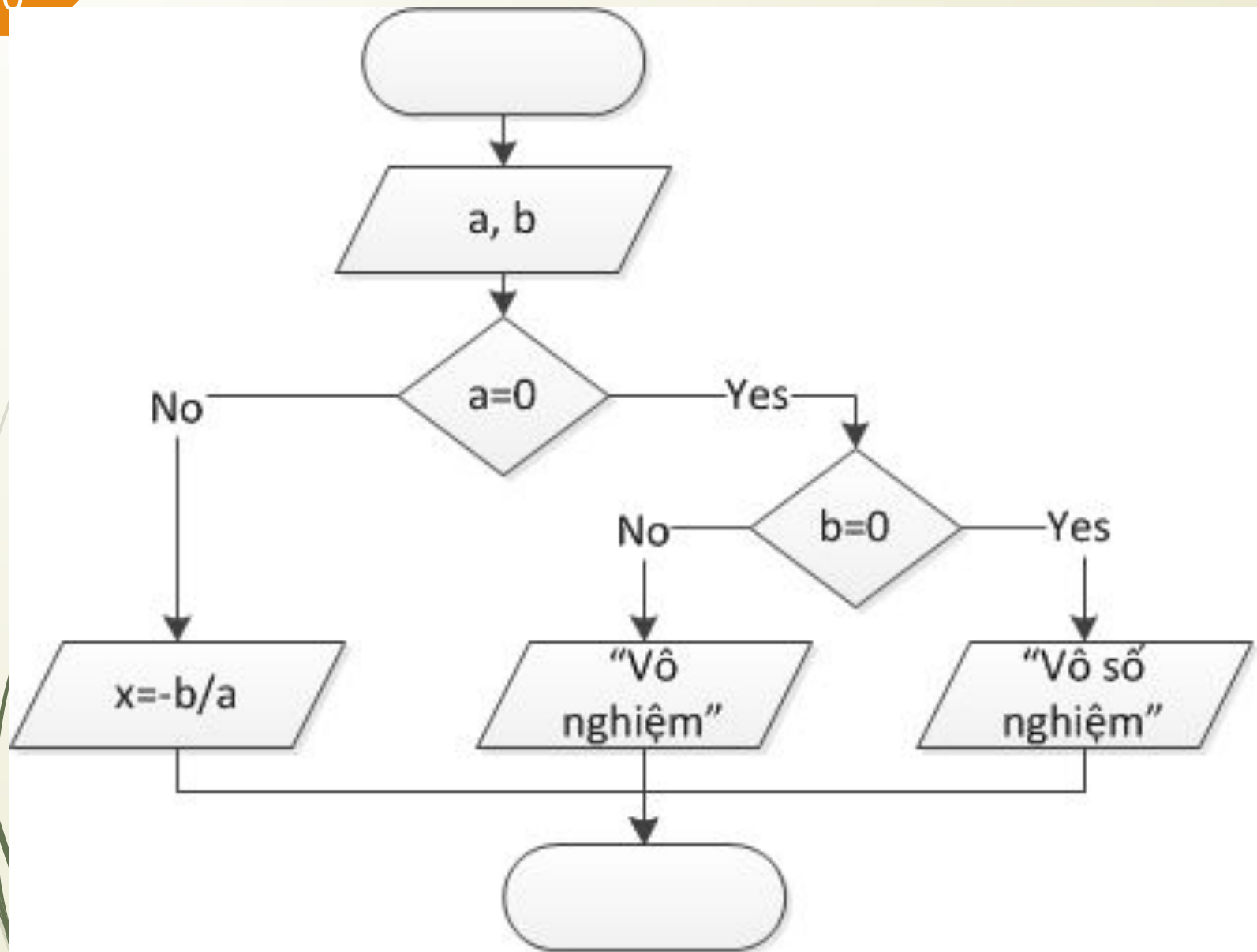
```
        Console.WriteLine (“a khong la boi so cua b”);
```

```
    }
```

```
}
```

## Ví dụ 2: Giải và biện luận phương trình: $ax+b=0$

10



```
static void Main(string []args)
```

```
{
```

```
    float a, b;
```

```
    Console.Write ("Nhap vao a: ");
```

```
    a = float.Parse (Console.ReadLine());
```

```
    Console.Write ("Nhap vao b: ");
```

```
    b = float.Parse (Console.ReadLine());
```

```
    if (a == 0)
```

```
    {
```

```
        if (b == 0)
```

```
        {
```

```
            Console.WriteLine ("PT VSN");
```

```
        }
```

```
        else
```

```
        {
```

```
            Console.WriteLine ("PT VN");
```

```
        }
```

```
    }
```

```
    else
```

```
    {
```

```
        Console.WriteLine ("Nghiem x = " + -b/a);
```

```
    }
```

```
}
```

## Bài tập viết chương trình

1. Nhập vào 2 số  $a$  và  $b$ , xuất ra số lớn nhất (Giả sử  $a \neq b$ )
2. Cho 3 số  $a$ ,  $b$  và  $c$ , hãy tìm giá trị lớn nhất của ba số trên và in ra kết quả (Giả sử  $a \neq b \neq c$ ).
3. Cho 3 số  $a$ ,  $b$  và  $c$ , hãy in ra màn hình theo thứ tự tăng dần các số (Giả sử  $a \neq b \neq c$ ).
4. Viết chương trình nhập vào một số nguyên  $n$  gồm 3 chữ số. Xuất ra màn hình chữ số lớn nhất ở vị trí nào? (Giả sử các chữ số có giá trị khác nhau)

VD:  $n=291$ . Chữ số lớn nhất nằm ở hàng chục (9).

5. Nhập vào số nguyên  $n$  gồm 3 chữ số. Xuất ra màn hình theo thứ tự tăng dần của các chữ số.

Ví dụ:  $n=291$ . Xuất ra 129.

6. Nhập vào ngày, tháng, năm. Kiểm tra xem ngày, tháng, năm đó có hợp lệ hay không? In kết quả ra màn hình.
7. Nhập vào giờ, phút, giây. Kiểm tra xem giờ, phút, giây đó có hợp lệ hay không? In kết quả ra màn hình.
8. Nhập vào một năm ( $>0$ ). Cho biết năm này có phải là năm nhuận hay không? In kết quả ra màn hình.

## 9. Tính tiền cước TAXI. Biết rằng:

14

*km đầu tiên là 13000<sup>d</sup>.*

*mỗi km tiếp theo là 12000<sup>d</sup>.*

*từ km 30 trở lên thì mỗi km thêm sẽ là 11000<sup>d</sup>.*

Hãy nhập số km sau đó in ra số tiền phải trả.

10. Nhập vào 3 số nguyên dương. Kiểm tra xem 3 số đó có lập thành tam giác không? Nếu có hãy cho biết tam giác đó thuộc loại nào? (Cân, vuông, đều, ...).
11. Nhập vào số nguyên dương  $n$ . Kiểm tra xem  $n$  có phải là số chính phương hay không? (số chính phương là số khi lấy căn bậc 2 có kết quả là nguyên).

12. Viết chương trình tính tiền thuê máy dịch vụ Internet và in ra màn hình kết quả. Với dữ liệu  
15 nhập vào là giờ bắt đầu thuê (GBD), giờ kết thúc thuê (GKT), số máy thuê (SoMay).

+ Điều kiện cho dữ liệu nhập:

$6 \leq GBD < GKT \leq 21$ . Giờ là số nguyên.

+ Đơn giá: 2500đ cho mỗi giờ máy trước 17 giờ và 3000đ cho mỗi giờ máy từ sau 17.

13. Viết chương trình tính tiền lương ngày cho công nhân, cho biết trước giờ vào ca, giờ ra ca của mỗi người.

Biết rằng:

+ Tiền trả cho mỗi giờ trước 12 giờ là 6000đ và sau 12 giờ là 7500đ. + Giờ vào ca sớm nhất là 6 giờ sáng và giờ ra ca trễ nhất là 18 giờ (Giả sử giờ nhập vào nguyên).

# Cấu trúc lựa chọn

16

**switch** (biểu thức)

{

**case** *n1*:  
    *các câu lệnh* ;  
    *break* ;

Trường hợp giá trị biểu  
thức bằng *n1*

**case** *n2*:  
    *các câu lệnh* ;  
    *break* ;

Trường hợp giá trị biểu  
thức bằng *n2*

.....

**case** *nk*:  
    <*các câu lệnh*> ;  
    *break* ;

[**default**: *các câu lệnh*]

Các trường hợp còn lại  
(nếu có)

}



Với:

- $n_i$  là các **hằng số nguyên hoặc ký tự**.
- Phụ thuộc vào giá trị của biểu thức viết sau **switch**, nếu:
  - Giá trị này =  $n_i$  thì thực hiện câu lệnh sau case  $n_i$ .
  - Khi giá trị biểu thức không thỏa tất cả các  $n_i$  thì thực hiện câu lệnh sau **default** nếu có, hoặc thoát khỏi câu lệnh **switch**.

Khi chương trình đã thực hiện xong câu lệnh của **case**  $n_i$  nào đó thì nó sẽ thực hiện luôn các lệnh thuộc **case** bên dưới nó mà không xét lại điều kiện (do các  $n_i$  được xem như các nhãn)

□ Vì vậy, để chương trình thoát khỏi lệnh **switch** sau khi thực hiện xong một trường hợp, ta dùng lệnh **break**.

**Ví dụ:** Nhập vào số nguyên n có giá trị từ 1 đến 5. In cách đọc của số đó ra màn hình.

19

```
static void Main(string []args)
```

```
{
```

```
    int n;
```

```
    Console.Write ("Nhap vao n ( $1 \leq n \leq 5$ ): ");
```

```
    n = int.Parse (Console.ReadLine());
```

```
    switch (n)
```

```
    {
```

```
        case 1: Console.WriteLine ("So mot");           break;
```

```
        case 2: Console.WriteLine ("So hai");           break;
```

```
        case 3: Console.WriteLine ("So ba");            break;
```

```
        case 4: Console.WriteLine ("So bon");
```

```
        break;
```

```
        case 5: Console.WriteLine ("So nam");           break;
```

```
        default : Console.WriteLine ("Chua xu ly");
```

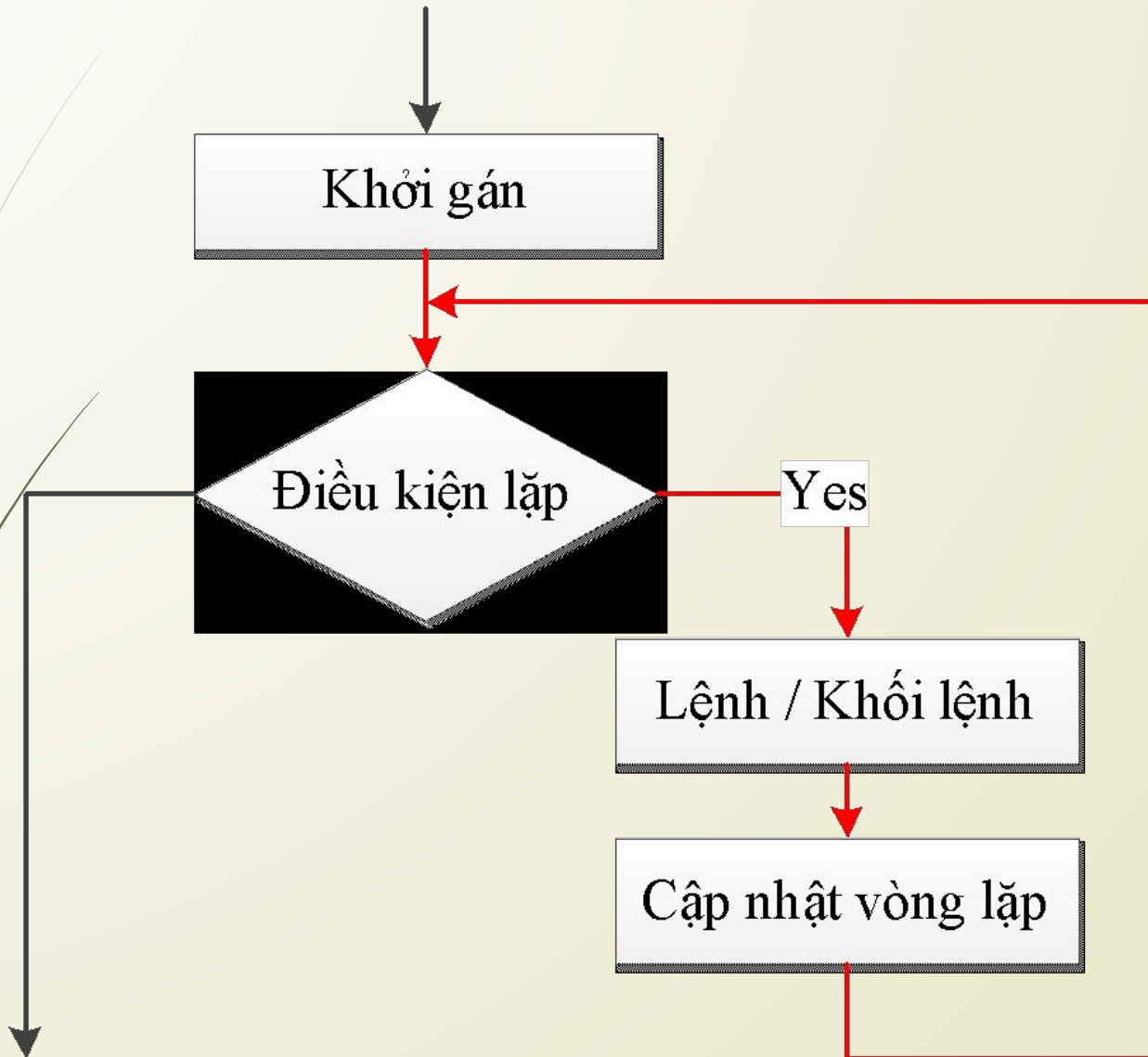
```
    }
```

```
}
```

## Bài tập viết chương trình

13. Viết chương trình nhập vào 1 số nguyên dương gồm 3 chữ số, in ra cách đọc của số này
14. Viết chương trình nhập vào ngày tháng năm, kiểm tra xem ngày tháng năm có hợp lệ không? In kết quả kiểm tra ra màn hình

# Cấu trúc lặp



# VÒNG LẶP *while*

*< Khởi gán >;*

***while*** (*<điều kiện lặp>*)

{

*lệnh/ khối lệnh;*

*<cập nhật>;*

}

*int k = 1;*

***while*** (*k <= 10*)

{

*Console.WriteLine(k);*

*k++;*

}

## Vòng lặp while

- Khởi gán: Dùng để khởi gán giá trị ban đầu cho vòng lặp
- Điều kiện lặp: Dùng để kiểm tra điều kiện trước khi thực hiện vòng lặp
- Cập nhật: Dùng để cập nhật vòng lặp (tăng hoặc giảm chỉ số lặp)

## Hoạt động của vòng lặp **while**

□ Bước 1: Khởi gán

□ Bước 2: Kiểm tra điều kiện

✓ Nếu điều kiện bằng *true* thì cho thực hiện các lệnh của vòng lặp, thực hiện **cập nhật vòng lặp**. Quay trở lại bước 2.

✓ Ngược lại thoát khỏi lặp.



Ví dụ: In ra màn hình 10 dòng chữ “*Xin chào*”

```
void main()
```

```
{
```

```
    int dong = 1;
```

```
    while(dong <= 10)
```

```
    {
```

```
        Console.WriteLine ("Xin chào ");
```

```
        dong++;
```

```
    }
```

```
}
```

Ví dụ: Nhập vào số nguyên dương  $n$ . In ra màn hình  $n$  dòng chữ “*Xin chào*”

```
void main()
```

```
{
```

```
    int n, dong = 1;
```

```
    Console.Write (“Nhập vào số nguyên dương n:”);
```

```
    n = int.Parse(Console.ReadLine());
```

```
    while(dong <= n)
```

```
    {
```

```
        Console.WriteLine("Xin chào");
```

```
        dong++;
```

```
    }
```

```
}
```

## Vòng lặp for

```
for (<khởi gán>;<điều kiện lặp>;<cập nhật>)  
{  
    <khối lệnh>;  
}
```

- **Lưu ý:** Cách hoạt động giống *while*

Ví dụ: In ra màn hình 10 dòng chữ “*Xin chào*”

28

```
void main()
```

```
{
```

```
    for (int dong = 1; dong <= 10; dong++)
```

```
    {
```

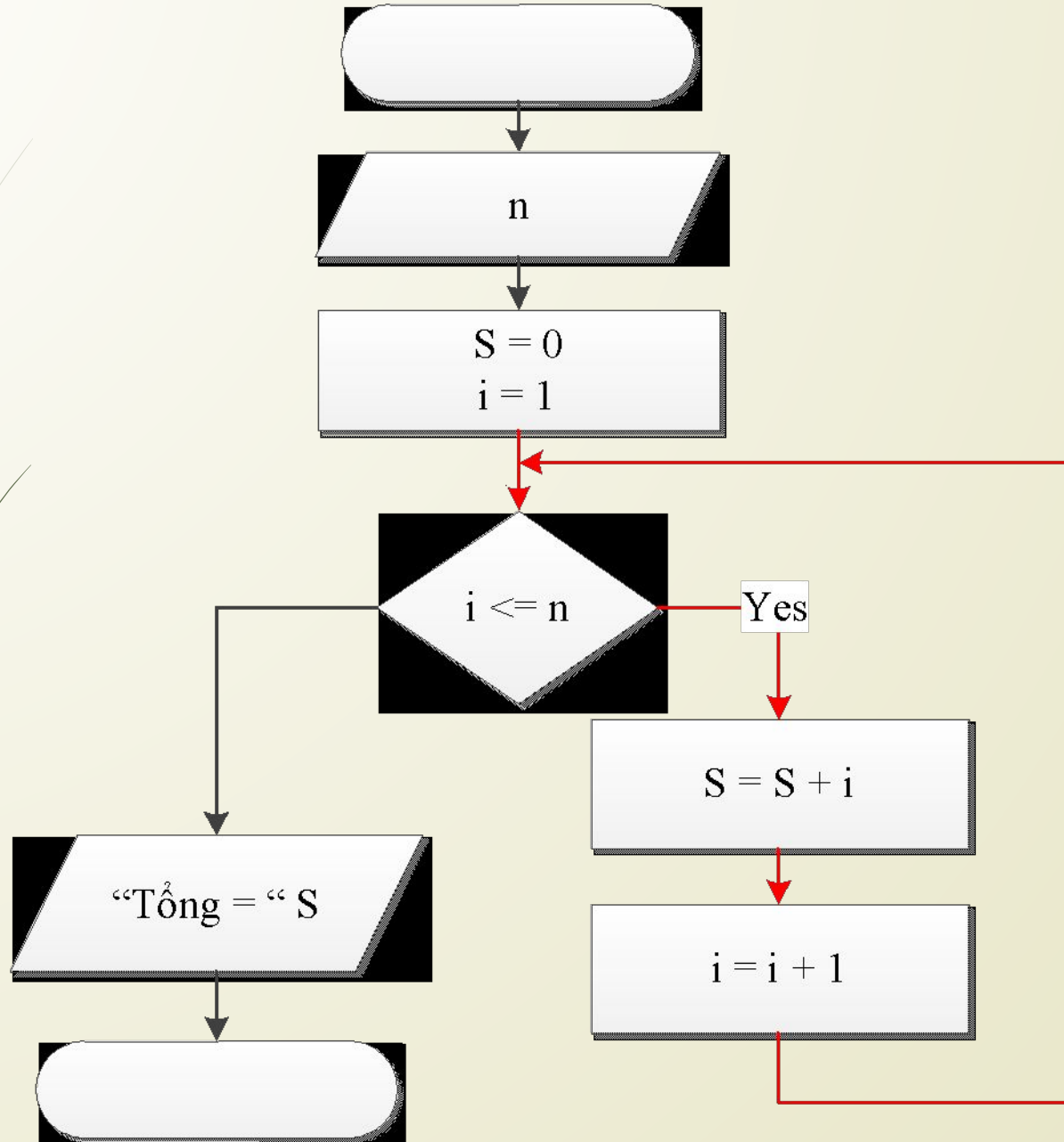
```
        Console.WriteLine(“Xin chào”);
```

```
    }
```

```
}
```

Tính tổng:  $S = 1 + 2 + 3 + \dots + n$ , với  $n > 0$

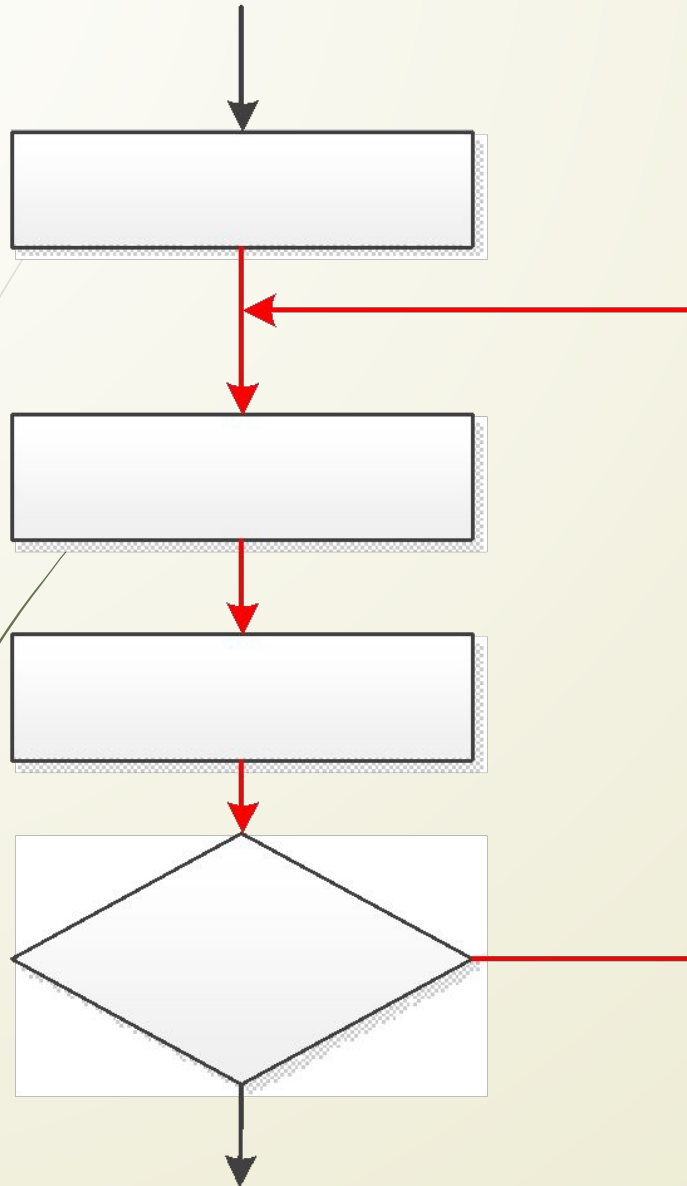
29



```
static void Main(string []args)
{
    int n, s=0;
    Console.Write ("Nhap so nguyen duong n: ");
    n = int.Parse(Console.ReadLine());
    for(int i=1; i<=n; i++)
    {
        s=s+i;
    }
    Console.WriteLine ("Tong = " + s);
}
```

# Vòng lặp do ... while

31



*<khởi gán>;*

**do**

{

*<khối lệnh>;*

*<cập nhật>;*

**} while** (điều kiện);

## Vòng lặp do ... while

- *Thực hiện khối lệnh cho đến khi biểu thức có giá trị bằng **false**.*
- *Cấu trúc lặp do...while thường được dùng cho trường hợp nhập dữ liệu có kiểm tra điều kiện*



**Ví dụ:** Nhập vào một số nguyên dương, nếu nhập sai thì thông báo lỗi và yêu cầu nhập lại.

33

```
static void Main(string []args)
```

```
{
```

```
    int n;
```

```
    do{
```

```
        Console.Write ("Nhap vao mot so nguyen duong: ");
```

```
        n = int.Parse(Console.ReadLine());
```

```
        if (n <= 0)
```

```
        {
```

```
            Console.WriteLine ("Nhap sai, hay nhap lai!");
```

```
        }
```

```
    } while (n <= 0);
```

```
    Console.WriteLine ("Ban da nhap dung, n = " + n);
```

```
}
```

□ Ngược lại với cấu trúc lặp *for* và *while* (kiểm tra điều kiện trước khi thực hiện lặp), vòng lặp *do...while* thực hiện lệnh lặp rồi mới kiểm tra điều kiện. Do đó vòng lặp *do...while* thực hiện lệnh ít nhất một lần.

# Lệnh break, return và continue

- Lệnh **break**: thoát khỏi các cấu trúc *switch*, *while*, *for*, *do...while*, *foreach* chứa nó
- Lệnh **return**: Kết thúc hàm
- Lệnh **continue**: bỏ qua các lệnh còn lại của vòng lặp

**VD:** Cho phép người dùng nhập liên tục số nguyên dương, nếu nhập số nguyên âm thì dừng

```
static void Main(string []args)
{
    int n;
    while (true)
    {
        Console.Write ("Nhap gia tri n: ");
        n = int.Parse(Console.ReadLine());
        if (n <= 0)
        {
            Console.WriteLine("Ket thuc vong lap");
            break;
        }
    }
}
```

VD: In ra màn hình giá trị từ 10 đến 20 trừ đi số 13 và số 17.

37

```
static void Main(string []args)  
{  
    for (int k = 10; k <= 20; k++)  
    {  
        if (k != 13 && k != 17)  
            Console.WriteLine(k);  
    }  
}
```

VD: In ra màn hình giá trị từ 10 đến 20 trừ đi số 13 và số 17 – Dùng *continue*

```
static void Main(string []args)  
{  
    for (int k = 10; k <= 20; k++)  
    {  
        if (k == 13 || k == 17)  
            continue;  
        Console.WriteLine(k);  
    }  
}
```

# Xác định kết quả

39

```
int a=18;  
for(int i = 1; i <= a; i++)  
    if(a%i == 0)  
        Console.WriteLine(i);
```

---

```
----  
for(int i = 0; i < 5; i++)  
{  
    for(int j = 0; j <= i; j++)  
        Console.Write(j + "\t");  
    Console.WriteLine();  
}
```

```
int i = 10, s = 0;
```

```
while(i > 0)
```

```
{
```

```
    if(i%2 == 0)
```

```
        s+=i;
```

```
    else
```

```
        if(i > 5)
```

```
            s+=2*i;
```

```
    i--;
```

```
}
```

```
Console.WriteLine("s = " + s);
```



```
int a = 18, i = 1;
```

```
do
```

```
{
```

```
    if(a%i == 0)
```

```
        Console.WriteLine(i + "\t");
```

```
    i++;
```

```
} while(i <= a);
```

```
int a = 11, b = 16, i = a;
```

```
while( i < b )
```

```
{
```

```
    if(i%2 == 0)
```

```
    {
```

```
        Console.Write(i + "\t");
```

```
        break;
```

```
    }
```

```
    i++;
```

```
}
```

```
int a = 10, s = 0, i = 0;
```

```
while( i < a )
```

```
{
```

```
    i++;
```

```
    if (i % 2 == 0)
```

```
        continue;
```

```
    s=s+i;
```

```
}
```

```
Console.WriteLine("s = " + s);
```

```
int i = 1, s = 0;
```

```
while(true)
```

```
{
```

```
    s = s + i++;
```

```
    if (i % 2)
```

```
        i = i + 2;
```

```
    else
```

```
        i = i + 1;
```

```
    if (i > 20)
```

```
        break;
```

```
}
```

```
Console.WriteLine("s = " + s);
```

15. Viết chương trình đếm số ước số của số nguyên dương  $N$ .

Ví dụ:  $N=12$

*số ước số của 12 là 6*

16. Nhập số nguyên dương  $n$ . Kiểm tra  $n$  có phải là số nguyên tố không?

17. Nhập vào 2 số nguyên dương  $a$  và  $b$ . Tìm ước số chung lớn nhất của  $a$  và  $b$ .

## 18. In ra dãy số Fibonacci

$$\square f_1 = f_0 = 1;$$

$$\square f_n = f_{n-1} + f_{n-2}; \quad (n > 1)$$

19. In hình chữ nhật có kích thước  $cd \times cr$

20. In hình tam giác vuông có chiều cao  $h$

