

# CẤU TRÚC ĐIỀU KHIỂN

- 1) Các ký hiệu thường gặp trong chương trình viết bằng ngôn ngữ C.

2) Các kiểu dữ liệu cơ bản, các phép toán, các hàm cơ bản trong C.

3) Các cấu trúc điều khiển: rẽ nhánh, lựa chọn, lặp.

4) Mô tả cách hoạt động và hướng dẫn chạy từng bước chương trình.

## I. LÝ THUYẾT

### 1. Các ký hiệu

STT	KÝ HIỆU	DIỄN GIẢI
1	{ }	Bắt đầu và kết thúc hàm hay khối lệnh.
2	;	Kết thúc khai báo biến, một lệnh, một lời gọi hàm, hay khai báo nguyên mẫu hàm.
3	//	Chú thích cho một dòng; chỉ có tác dụng đối với người đọc chương trình.
4	/* */	Tương tự như dấu //, nhưng cho trường hợp nhiều dòng.

### 2. Các kiểu dữ liệu cơ bản trong C

STT	KIỂU	Ý NGHĨA	KÍCH THƯỚC	ĐỊNH DẠNG
KIỂU SỐ THỰC				
1	float		4 bytes	%f
2	double		8 bytes	%lf
3	long double		10 bytes	%lf
KIỂU SỐ NGUYÊN				
1	char	Ký tự	1 bytes	%c
		Số nguyên	1 bytes	%d
2	unsigned char	Số nguyên	1 bytes	%d
3	int	Nguyên dương	2 bytes	%d
4	unsigned int	Số nguyên	2 bytes	%u
5	long	Nguyên dương	4 bytes	%ld
6	unsigned long	Số nguyên	4 bytes	%lu
7	char*	Chuỗi		%s

### 3. Các phép toán trong C

STT	PHÉP TOÁN	Ý NGHĨA	GHI CHÚ
-----	-----------	---------	---------

PHÉP TOÁN SỐ HỌC			
1	+	Cộng	
2	-	Trừ	
3	*	Nhân	
4	/	Chia lấy phần nguyên	
5	%	Chia lấy phần dư	
PHÉP TOÁN QUAN HỆ			
1	>	Lớn hơn	
2	<	Nhỏ hơn	
3	>=	Lớn hơn hoặc bằng	
4	<=	Nhỏ hơn hoặc bằng	
5	= =	Bằng nhau	
6	!=	Khác nhau	
PHÉP TOÁN LOGIC			
1	!	NOT	Giá trị của hạng bằng 0 hay khác 0.
2	&&	AND	
3		OR	
PHÉP TOÁN THAO TÁC VỚI BIT			
1	&	AND	
2		OR	
3	^	XOR	
4	<<	Dịch trái	
5	>>	Dịch phải	
6	~	Lấy phần bù theo bit	
PHÉP TOÁN THAO TÁC VỚI BIT			
1	=	Gán	Có += -= *= /= %= <<= >>= &=  = ^=
2	++	Tăng 1	Nếu toán tử tăng/giảm đặt trước thì tăng/giảm trước rồi tính biểu thức hoặc ngược lại.
3	--	Giảm 1	

#### 4. Các hàm cơ bản trong C

STT	TÊN HÀM	THƯ VIỆN	DIỄN GIẢI
1	<b>printf</b>	#include<stdio.h>	Xuất ra màn hình.
2	<b>scanf</b>	#include<stdio.h>	Lấy dữ liệu từ bàn phím.
3	<b>gotoxy</b>	#include<conio.h>	Di chuyển dấu nháy đến toạ độ (x,y) trên màn hình văn bản.
4	<b>textcolor</b>	#include<conio.h>	Đặt màu cho chữ (giá trị 0÷15).

5	<b>cprintf</b>	#include<stdio.h>	Xuất ra màn hình với màu chữ đã định liền trước đó.
6	<b>delay</b>	#include<dos.h>	Dừng thực hiện lệnh tiếp sau một khoảng thời gian.
7	<b>kbhit</b>	#include<conio.h>	Kiểm tra xem có nhấn phím.

## 5. Cấu trúc rẽ nhánh

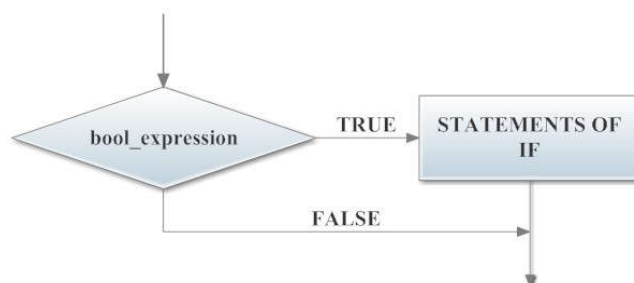
### a. Cấu trúc if

```

if (biểu thức điều kiện)
{
    <khối lệnh>;
}

```

Nếu biểu thức điều kiện cho kết quả khác không thì thực hiện khối lệnh



**Hình 2.1:** Hoạt động của cấu trúc if

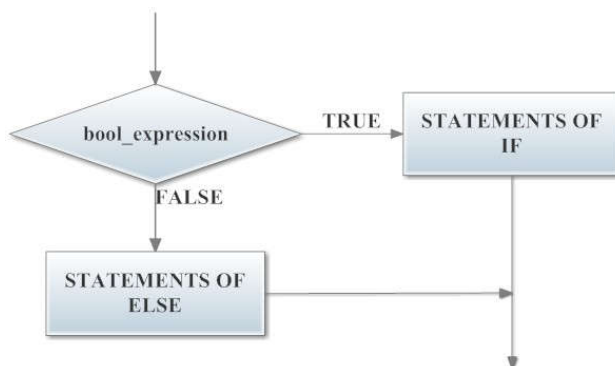
### b. Cấu trúc if..... else

```

if (<biểu thức điều kiện>)
{
    <khối lệnh if>;
}
else
{
    <khối lệnh else>;
}

```

Nếu biểu thức điều kiện cho kết quả khác không thì thực hiện khối lệnh if, ngược lại thì cho thực hiện khối lệnh else. Biểu thức điều kiện phải đặt trong cặp dấu ngoặc đơn.



**Hình 2.2:** Hoạt động của cấu trúc if..... else

## 6. Cấu trúc lựa chọn switch

```

switch (biểu thức)

```

```

{
  case  $n_1$ :
    <các câu lệnh>;
    break;

  case  $n_2$ :
    <các câu lệnh>;
    break;

  case  $n_k$ :
    <các câu lệnh>;
    break;

  default:
    <các câu lệnh>;
}

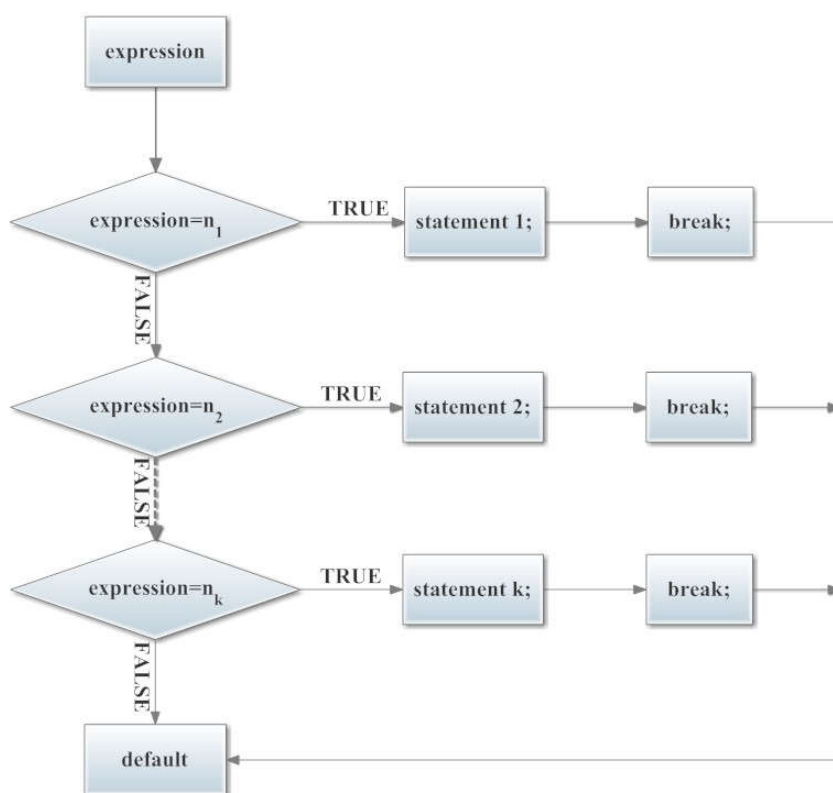
```

$n_i$  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**.



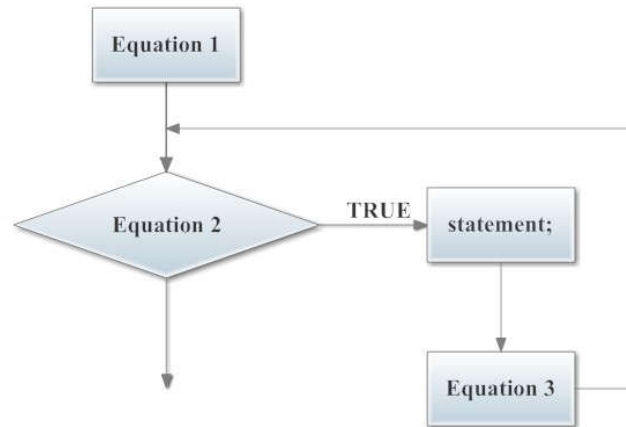
**Hình 2.3:** Hoạt động của cấu trúc **switch**

## 7. Cấu trúc lặp

### a. Cấu trúc *for*

```
for (<biểu thức khởi gán>;<biểu thức điều kiện>;<biểu thức tăng/giảm>)
{
    <khối lệnh>;
}
```

Bất kỳ biểu thức nào trong 3 biểu thức trên đều có thể vắng mặt nhưng phải có dấu (;).

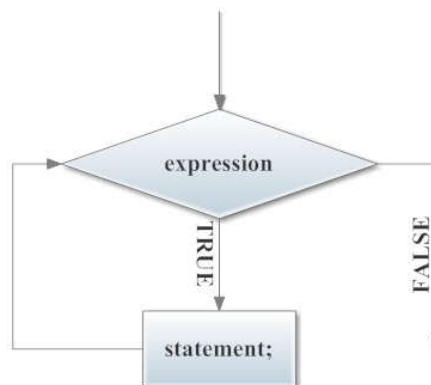


**Hình 2.4:** Hoạt động của cấu trúc *for*

### b. Cấu trúc *while*

```
<khởi gán>
while (<biểu thức điều kiện>)
{
    <khối lệnh>;
    <tăng/giảm chỉ số lặp>;
}
```

Cách hoạt động của cấu trúc **while** giống như cấu trúc **for** nếu ta có khởi gán trước và tăng/giảm chỉ số lặp trong vòng lặp **while**.



**Hình 2.5:** Hoạt động của cấu trúc *while*

### c. Cấu trúc do ..... *while*

```
do
{
    <khối lệnh>;
}
```

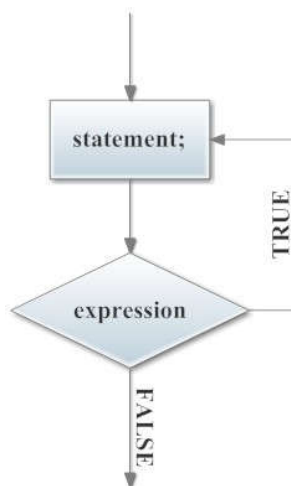
```

}
while (<biểu thức điều kiện>);

```

Thực hiện khối lệnh cho đến khi biểu thức có giá trị bằng 0.

**Lưu ý:** Lặp **while** kiểm tra điều kiện trước khi thực hiện lặp, còn 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.



**Hình 2.6:** Hoạt động của cấu trúc **do ..... while**

## 8. Lệnh break và continue

### a. Lệnh break

Dùng để kết thúc vòng lặp trực tiếp chứa nó khi thỏa điều kiện nào đó.

### b. Lệnh continue

Dùng để bỏ qua một lần lặp khi thỏa điều kiện nào đó.

## II. NỘI DUNG THỰC HÀNH

### 1. Cấu trúc rẽ nhánh if, lựa chọn switch

**Viết các chương trình theo thứ tự sau đây**

- a. Nhập vào số nguyên n. Kiểm tra nếu  $n > 0$  tăng lên 1 đơn vị. Xuất kết quả ra màn hình.
- b. Nhập vào số nguyên n. Kiểm tra:
  - Nếu  $n > 0$  tăng lên 1 đơn vị
  - Nếu  $n < 0$  thì giảm xuống 1 đơn vị

Xuất kết quả ra màn hình

- c. Nhập vào số nguyên dương n, giá trị từ  $0 \div 10$ . Kiểm tra:
  - Nếu  $0 \leq n < 3$  thì kết quả là Kém.
  - Nếu  $3 \leq n < 5$  thì kết quả là Yếu.
  - Nếu  $5 \leq n < 6$  thì kết quả là Trung bình.

- Nếu  $6 \leq n < 8$  thì kết quả là Khá.
- Nếu  $8 \leq n < 9$  thì kết quả là Giỏi.
- Nếu  $9 \leq n \leq 10$  thì kết quả là Xuất sắc.
- Các trường hợp còn lại là Sai.

## 2. Cấu trúc lặp for, while

**Viết các chương trình sau, sử dụng đồng thời các cấu trúc for, while, do ..... while rồi so sánh**

- a. Nhập vào số nguyên n. Xuất kết quả ra màn hình các số từ 1 đến n.
- b. Nhập vào số nguyên n. Tính:
  - $1+2+3+\dots+n$
  - $n!$
  - $3^n$

## III. BÀI TẬP

### Cấu trúc if; if ..... else và switch

1. Cho biết kết quả của chương trình sau:

```
int a=9, b=6;
a++;
a=a+b--;
a=a+ (--b);
if(a%2==0)
    printf("Gia tri cua a la chan");
printf("Tong cua a va b la: %d", a+b);
```

2. Cho biết kết quả của đoạn chương trình sau:

```
int a=7, b=8;
a++;
a=a+(b--);
--b;

a--;
a=(--a)+(--b);
if(a%2!=0)
    printf("\n a la so le");
else
    printf("\n a la so chan"); printf("\na = %d",a);
```

3. Cho biết kết quả của đoạn chương trình sau:

```
int x=5, y;
y=x++ + 5;
printf("x=%d, y=%d\n", x, y); y*=6;
x=y%7;
printf("x=%d, y=%d, y/x=%d", x, y, y/x);
```

4. Nhập vào hai số nguyên a, b. In ra màn hình giá trị lớn nhất.

5. Cho ba số a, b, c đọc vào từ bàn phím. Hãy tìm giá trị lớn nhất của ba số trên và in ra kết quả.
6. Cho ba số a, b, c đọc vào từ bàn phím. Hãy in ra màn hình theo thứ tự tăng dần các số. (Chỉ được dùng thêm hai biến phụ).
7. Viết chương trình nhập vào một số nguyên n gồm ba chữ số. Xuất ra màn hình chữ số lớn nhất ở vị trí nào?  
Ví dụ: n=291. Chữ số lớn nhất nằm ở hàng chục (9).
8. Viết chương trình nhập vào số nguyên n gồm ba chữ số. Xuất ra màn hình theo thứ tự tăng dần của các chữ số. Ví dụ: n=291. Xuất ra 129.
9. Nhập vào ngày, tháng, năm. Kiểm tra xem ngày, tháng, năm đó có hợp lệ hay không? In kết quả ra màn hình.
10. Nhập vào giờ, phút, giây. Kiểm tra xem giờ, phút, giây đó có hợp lệ hay không? In kết quả ra màn hình.
11. Viết chương trình nhập vào ngày, tháng, năm hợp lệ. Cho biết năm này có phải là năm nhuận hay không? In kết quả ra màn hình.
12. Viết chương trình tính diện tích và chu vi các hình: tam giác, hình vuông, hình chữ nhật và hình tròn với những thông tin cần được nhập từ bàn phím.
13. Viết chương trình tính tiền cước TAXI. Biết rằng:
  - o KM đầu tiên là 12.500đ.
  - o 200m tiếp theo là 2.000đ.
  - o Nếu lớn hơn 30km thì mỗi km thêm sẽ là 9.000đ.

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

14. Nhập vào 3 số nguyên dương a, b, c. 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,...).
15. Viết chương trình 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).

### Cấu trúc lặp

16. Cho biết kết quả của đoạn chương trình sau:

```
int a=18;
for(int i=1; i<=a; i++)
    if(a%i==0)
        printf("\t %d", i);
```

17. Cho biết kết quả của đoạn chương trình sau:

```
for(int i=0; i<5; i++)
{
    for(int j=0; j<=i; j++)
        printf("%d\t", j);
    printf("\n");
}
```



18. Cho biết kết quả của đoạn chương trình sau:

```
int i=10, s=0;
while(i>0)
{
    if(i%2==0)
        s+=i;
    else
        if(i>5)
            s+=2*i;
            i--;
}
printf("s = %d",s);
```

19. Cho biết kết quả của đoạn chương trình sau:

```
int a=18, i=1;
do
{
    if(a%i==0)
        printf("\t %d",i);
    i++;
}
while(i<=a);
```

20. Cho biết kết quả của đoạn chương trình sau:

```
int a=11, b=16, i=a;
while( i<b )
{
    if(i%2==0)
    {
        printf("\t %d", i);
        break;
    }
    i++;
}
```

21. Cho biết kết quả của đoạn chương trình sau:

```
int a=10, s=0, i=0;
while( i<a )
{
    i++;
    if(i%2==0) continue;
    else s=s+i;
}
printf("s=%d",s);
```

22. Cho biết kết quả của đoạn chương trình sau:

```
int i=1,s=0;
while(1)
{
    s=s+i++;
    if(i%2)
        i=i+2;
```

```

else
    i=i+1;
if(i>20)
    break;
}
printf("%d",s);

```

23. Viết chương trình in ra màn hình hình chữ nhật đặc, kích thước  $m \times n$  ( $m, n$  nhập từ bàn phím).

Ví dụ: Nhập  $m=5, n=4$

```

* * * * *
* * * * *
* * * * *
* * * * *

```

24. Viết chương trình in ra màn hình hình chữ nhật rỗng, kích thước  $m \times n$  ( $m, n$  nhập từ bàn phím).

Ví dụ: Nhập  $m=5, n=4$

```

* * * * *
*           *
*           *
* * * * *

```

25. Viết chương trình in ra màn hình tam giác vuông cân đặc, có độ cao  $h$  ( $h$  nhập từ bàn phím).

Ví dụ: Nhập  $h=4$

```

*
* *
* * *
* * * *

```

26. Viết chương trình in ra màn hình tam giác cân rỗng có độ cao  $h$  ( $h$  nhập từ bàn phím).

Ví dụ: Nhập  $h=4$

```

*
* *
*   *
* * * *

```

27. Viết chương trình in ra màn hình tam giác cân đặc có độ cao  $h$  ( $h$  nhập từ bàn phím).

Ví dụ: Nhập  $h=4$

```

      *
     * * *
    * * * * *
   * * * * *
  * * * * *

```

28. Viết chương trình in ra màn hình tam giác cân rỗng có độ cao  $h$  ( $h$  nhập từ bàn phím).

Ví dụ: Nhập  $h=4$

```

      *
    *   *
  *       *
* *   * * *

```

29. Viết chương trình nhập số nguyên dương n. Liệt kê n số nguyên tố đầu tiên.
30. Viết chương trình nhập vào hai số nguyên dương a và b. Tìm ước số chung lớn nhất và bội số chung nhỏ nhất của a và b.
31. Viết chương trình nhập vào một số nguyên n gồm tối đa 10 chữ số (4 bytes). In ra màn hình giá trị nhị phân của số trên. (Hướng dẫn: chia lấy dư cho 2 và xuất theo thứ tự ngược lại dùng hàm gotoxy, wherex, wherey).
32. Viết chương trình đếm số ước số của số nguyên dương N. Ví dụ: N=12 số ước số của 12 là 6
33. Một số hoàn thiện là một số có tổng các ước số của nó (không kể nó) bằng chính nó. Hãy liệt kê các số hoàn thiện nhỏ hơn 5000.

Ví dụ: số 6 là số hoàn thiện vì tổng các ước số là  $1+2+3=6$ .

34. Nhập vào ngày, tháng, năm. Cho biết đó là ngày thứ mấy trong năm.

In ra dãy số Fibonacci

$$f_1 = f_0 = 1;$$

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