

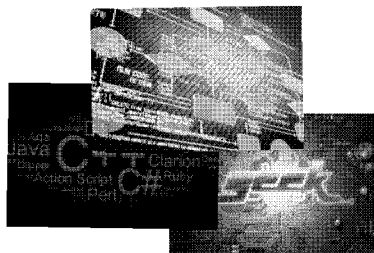


ĐẠI HỌC QUỐC GIA TP HCM
TRƯỜNG ĐẠI HỌC
CÔNG NGHỆ THÔNG TIN

NHẬP MÔN LẬP TRÌNH

CHƯƠNG IV

CÁC CẤU TRÚC ĐIỀU KHIỂN



Nguyễn Trọng Chính
chinhnt@uit.edu.vn

CÁC CẤU TRÚC ĐIỀU KHIỂN

- ❖ KHÁI NIỆM
- ❖ CẤU TRÚC Rẽ NHÁNH
- ❖ CẤU TRÚC LẶP
- ❖ BREAK VÀ CONTINUE

KHÁI NIỆM

❖CẤU TRÚC ĐIỀU KHIỂN

Cấu trúc điều khiển là một cấu trúc cho phép thay đổi trình tự thực hiện các câu lệnh trong chương trình.

❖CÂU LỆNH (statement)

Trong C/C++, mỗi câu lệnh được kết thúc bằng dấu ;

❖KHỐI LỆNH

Còn được gọi là lệnh ghép, là một tập hợp các câu lệnh được đặt trong cặp dấu { và }. Khối lệnh có thể xem như là một câu lệnh.

3

KHÁI NIỆM

Ví dụ:

Câu lệnh:

```
int a, b = 5, c = 2;
```

```
a = b + c++;
```

```
(a % c) ? printf("%d", a*a) : printf("%d", a/2);
```

khối lệnh:

```
{
```

```
int a, b;
```

```
scanf("%d %d", &a, &b);
```

```
...
```

```
}
```

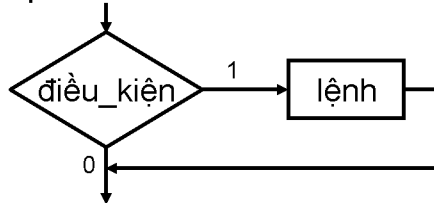
4

CẤU TRÚC Rẽ NHÁNH

❖CẤU TRÚC IF

* **Dạng khuyết:** cho phép thực hiện một câu lệnh nếu thỏa điều kiện xác định.

if (điều_kiện)
lệnh



trong đó,

- **điều_kiện** là một biểu thức mang ý nghĩa logic.
- **lệnh** là một câu lệnh. Trong trường hợp cần thực hiện nhiều lệnh nếu thỏa điều kiện thì các lệnh đó được đưa vào một khối lệnh

5

CẤU TRÚC Rẽ NHÁNH

❖CẤU TRÚC IF

Ví dụ

- 1) Nhập vào 2 số nguyên a và b, in ra kết quả phép chia như sau:
 - nếu a chia hết cho b thì in “a chia b duoc x”
 - ngược lại thì in “a chia b duoc x du y”
- 2) Nhập vào tên, năm sinh của một người. Nếu tuổi của người đó lớn hơn 14 thì nhập số CMND, ngày cấp. In ra thông tin của người đó.

6

CẤU TRÚC RỄ NHÁNH

1)

```
int a, b, r;  
scanf("%d %d", &a,&b);  
r = a % b;  
printf("%d chia %d duoc %d", a, b, a/b);  
if (r)  
    printf(" du %d\n", r);
```

7

CẤU TRÚC RỄ NHÁNH

2)

```
int ns, n = 2014;  
char ten[30], cmnd[11], nc[11];  
gets(ten); scanf("%d%*c", &ns);  
if ((n - ns) > 14) {  
    gets(cmnd); gets(nc);  
}  
printf("Ten: %s. \tsinh nam: %d.\n", ten,ns);  
if ((n - ns) > 14)  
    printf("So CMND: %s.\tNgay cap: %s.\n", cmnd, nc);
```

8

CẤU TRÚC RỄ NHÁNH

Lưu ý:

- Hàm **scanf** khi lấy giá trị cho biến không lấy ký tự `\n` trong dữ liệu nhập từ bàn phím. Để hủy các ký tự `\n` này sau khi nhập dữ liệu vào biến, thêm định dạng `%c` vào cuối chuỗi định dạng của hàm **scanf**.
- Hàm **gets** nhập chuỗi đến khi nào nhận được ký tự `\n`, nên phải xác định kích thước chuỗi có thể chứa được nội dung cần nhập. Hàm **gets** lấy ký tự `\n` trong dữ liệu nhập từ bàn phím nhưng không đưa vào biến.

9

CẤU TRÚC RỄ NHÁNH

❖CẤU TRÚC IF

* **Dạng đủ**: cho phép thực hiện một trong hai câu lệnh theo điều kiện xác định. Nếu thỏa thực hiện lệnh thứ nhất, ngược lại thực hiện lệnh thứ 2

if (điều_kiện)

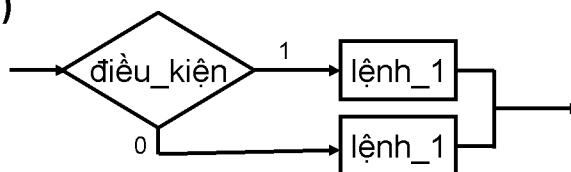
lệnh_1

else

lệnh_2

trong đó,

- **điều_kiện** là một biểu thức mang ý nghĩa logic.
- **lệnh_1, lệnh_2** là một câu lệnh như trong dạng khuyết.



10

CẤU TRÚC RỄ NHÁNH

❖CẤU TRÚC IF

Ví dụ

3) Viết chương trình nhập vào các số số nguyên ngày, tháng, năm và cho biết ngày đó có hợp lệ hay không.

11

CẤU TRÚC RỄ NHÁNH

3)

```
int nn, t, n;  
scanf("%d %d %d", &nn,&t,&n);  
if ((nn<1) || (t<1) || (n<1) || (nn>31) || (t>12))  
    printf("ngay khong hop le\n");  
else  
    if ((t==4) || (t==6) || (t==9) || (t==11))  
        if (nn < 31) printf("ngay hop le\n");  
        else printf("ngay khong hop le\n");  
    else  
        if (t==2)  
            if ((n%4==0) && (n%100 !=0) || (n%400 == 0))  
                if (nn < 30) printf("ngay hop le\n");  
                else printf("ngay khong hop le\n");
```

12

CẤU TRÚC RỄ NHÁNH

3)

```
else
    if (nn < 29) printf("ngay hop le\n");
    else printf("ngay khong hop le\n");
else
    printf("ngay hop le\n");
```

13

CẤU TRÚC RỄ NHÁNH

❖CẤU TRÚC SWITCH

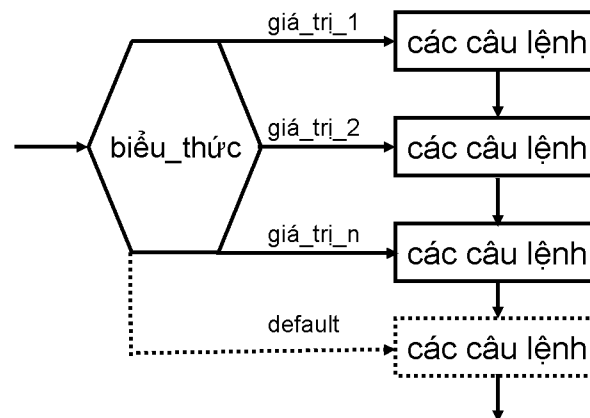
Cho phép thực hiện các câu lệnh nếu biểu thức có kiểu số nguyên thỏa một giá trị trong số các giá trị số nguyên xác định được liệt kê. Khi đó, tất cả câu lệnh bắt đầu tại vị trí giá trị thỏa biểu thức đến hết cấu trúc switch đều được thực hiện. Các câu lệnh tại giá trị default, nếu có, sẽ luôn được thực hiện.

```
switch (biểu_thức) {
    case giá_trị_1: các câu lệnh
    case giá_trị_2: các câu lệnh
    ...
    [default: các câu lệnh]
}
```

14

CẤU TRÚC RỄ NHÁNH

❖CẤU TRÚC SWITCH



15

CẤU TRÚC RỄ NHÁNH

❖CẤU TRÚC SWITCH

Ví dụ

- 4) Nhập vào một số nguyên cho biết kết quả học tập của học sinh và in ra các công việc sẽ được thực hiện, gồm:
- 1(trung bình): được lên lớp.
 - 2(khá): được lên lớp, nhận giấy khen.
 - 3(giỏi): được lên lớp, nhận giấy khen, nhận phần thưởng của trường.
 - 4(xuất sắc): được lên lớp, nhận giấy khen, nhận phần thưởng của trường, nhận phần thưởng của sở.

16

CẤU TRÚC RỄ NHÁNH

3)

```
int kq;  
scanf("%d", &kq);  
switch(kq) {  
    case 4: printf("nhan phan thuong cua so\n");  
    case 3: printf("nhan phan thuong cua truong\n");  
    case 2: printf("nhan giay khen\n");  
    case 1: printf("duoc len lop\n");  
    default: printf("=====\n");  
}
```

17

CẤU TRÚC LẶP

❖CẤU TRÚC FOR

Cho phép thực hiện một câu lệnh nhiều lần từ lúc khởi đầu đến khi giá trị của một biểu thức không còn thỏa điều kiện, cuối mỗi lần lặp lệnh sẽ thực hiện một biểu thức xác định.

**for (biểu_thức_1; điều_kiện; biểu_thức_2)
lệnh**

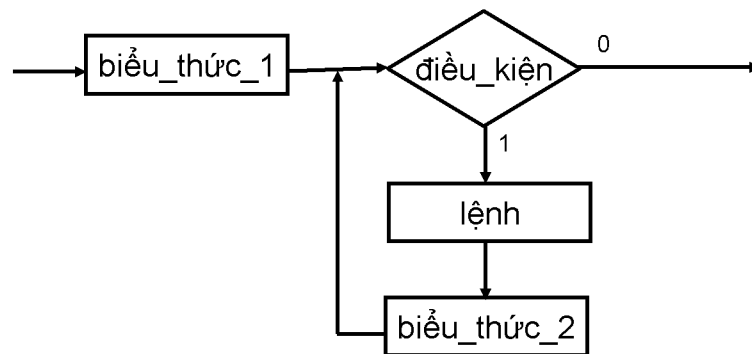
Trong đó,

- **biểu_thức_1** được thực hiện 1 lần khi bắt đầu.
- **điều_kiện** được xét trước mỗi lần lặp
- **biểu_thức_2** được thực hiện cuối mỗi lần lặp.
- **lệnh** là một lệnh hoặc một khối lệnh.

18

CẤU TRÚC LẶP

❖CẤU TRÚC FOR



19

CẤU TRÚC LẶP

❖CẤU TRÚC FOR

Ví dụ

5) Nhập vào 1 số nguyên dương n và tính

$$s=2+4+\dots+2k, 2k \leq n.$$

6) Nhập vào 1 số nguyên và in ra bảng cửu chương tại giá trị số nguyên đó

20

CẤU TRÚC LẶP

```
5)
int n, i, kq;
scanf("%d", &n);
kq = 0;
for (i = 2; i <= n; i += 2)
    kq = kq + i;
```

21

CẤU TRÚC LẶP

```
6)
int n, i;
for (printf("n = "),scanf("%d",&n),i=1; i<10;
    printf("%d x %d = %d\n",n,i,n*i),i++);
printf("=====\n");
```

Lưu ý: Nếu chỉ có dấu ; thì vẫn được xem là một lệnh nhưng không làm gì.

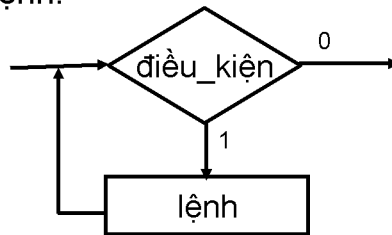
22

CẤU TRÚC LẶP

❖CẤU TRÚC WHILE

Cho phép thực hiện một lệnh nhiều lần nếu điều kiện xác định được thỏa. Điều kiện sẽ được kiểm tra trước khi thực hiện lệnh.

while (điều_kiện)
lệnh



Trong đó,

- **điều_kiện** là biểu thức mang ý nghĩa logic.
- **lệnh** có thể là một câu lệnh hoặc một khối lệnh

23

CẤU TRÚC LẶP

❖CẤU TRÚC WHILE

Ví dụ

7) nhập vào một số nguyên n và tính n!

24

CẤU TRÚC LẶP

```
7)
int n, kq = 1;
scanf("%d",&n);
while (n > 0) {
    kq = kq * n;
    n -=1;
}
```

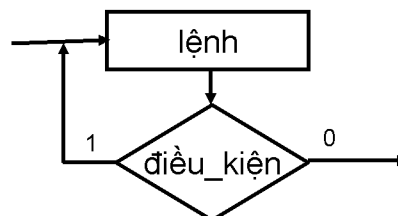
25

CẤU TRÚC LẶP

❖CẤU TRÚC DO WHILE

Cho phép thực hiện một lệnh nhiều lần nếu điều kiện xác định được thỏa. Điều kiện sẽ được kiểm tra sau khi thực hiện lệnh.

do
 lệnh
while (điều_kiện);



Trong đó,

- **điều_kiện** là biểu thức mang ý nghĩa logic.
- **lệnh** có thể là một câu lệnh hoặc một khối lệnh

26

CẤU TRÚC LẶP

❖CẤU TRÚC WHILE

Ví dụ

8) Viết chương trình đảm bảo nhập số nguyên dương có 3 chữ số và in ra số đảo ngược của nó.

27

CẤU TRÚC LẶP

```
8)
int n, dv, c, t;
do {
    scanf("%d",&n);
    t = d / 100;
} while ((t < 0) || (n<0) || (n > 999));
c = (n % 100) / 10;
dv = n % 10;
printf("%d%d%d",dv,c,t);
```

28

BREAK VÀ CONTINUE

❖CÂU LỆNH BREAK

Được dùng để thoát khỏi một cấu trúc rẽ nhánh ngay sau câu lệnh break mà không thực hiện bất kỳ lệnh nào sau nó trong cấu trúc chứa nó. Sử dụng:

break;

❖CÂU LỆNH CONTINUE

Được sử dụng để tiếp tục cấu trúc lặp ngay sau câu lệnh continue mà không thực hiện bất kỳ lệnh nào trong cấu trúc lặp chứa nó. Sử dụng:

continue;

29

BREAK VÀ CONTINUE

Ví dụ

9) Viết chương trình nhập vào một số từ 0 đến 9, in ra giá trị bằng chữ của số đó.

10) Viết chương trình cho phép thực hiện nhiều lần việc giải phương trình bậc hai $ax^2 + bx + c = 0$ đến khi nhập $a = b = c = 0$.

30

BREAK VÀ CONTINUE

9)

```
int n;
scanf("%d",&n);
switch (n) {
    case 0: printf("khong\n"); break;
    case 1: printf("mot\n"); break;
    case 2: printf("hai\n"); break;
    case 3: printf("ba\n"); break;
    case 4: printf("bon\n"); break;
    case 5: printf("nam\n"); break;
    case 6: printf("sau\n"); break;
    case 7: printf("bay\n"); break;
    case 8: printf("tam\n"); break;
    case 9: printf("chin\n"); break;
}
```

31

BREAK VÀ CONTINUE

10)

```
double a, b, c, d, x1, x2;
while (1) {
    scanf("%f %f %f",&a, &b, &c);
    if ((a==0) && (b==0) && (c==0))
        break;
    if (a==0)
        continue;
    d = b*b - 4*a*c;
    if (d < 0) {
        printf("pt vo nghiem\n"); continue;
    }
    d = sqrt(d); a = 2*a; x1 = (-b + d)/a; x2 = (-b-d)/a;
    printf("pt co nghiem x1 = %f, x2 = %f\n", x1, x2);
}
```

32

BÀI TẬP

- 1) Viết chương trình nhập vào một số từ 0 đến 999, in ra giá trị bằng chữ của số đó.
- 2) Viết chương trình nhập vào một số nguyên dương n và in ra tất cả số nguyên tố nhỏ hơn n .
- 3) Viết chương trình nhập vào một số tiền t và in ra tất cả các phương án dùng số lượng tờ của các loại tiền 1đ, 2đ, 5đ, 10đ, 20đ, 50đ để có giá trị bằng t .
- 4) Viết chương trình nhập vào một số n và in ra tất cả số chính phương có bình phương không quá n .