



2.4 Các lệnh cơ bản (tiếp)

Nội dung

- Lệnh khối
- Lệnh lặp
- Lệnh break và continue
- Lệnh điều khiển rẽ nhánh



Lệnh khối

2.3 Các lệnh cơ bản (tiếp)

- **Lệnh khối:** là dãy các câu lệnh được đặt trong cặp dấu ngoặc nhọn { }.

```
{  
    lệnh_1;  
    lệnh_2;  
    ....  
    lệnh_n;  
}
```

Lệnh khối

- Trong lệnh khối có thể chứa lệnh khối khác, ta gọi đó là các lệnh khối lồng nhau. Sự lồng nhau của các lệnh khối là không hạn chế.

```
{
    lệnh;
    {
        lệnh;
        ...
    }
    ...
}
```

Lệnh khối

- C cho phép khai báo biến trong lệnh khối. Ràng buộc duy nhất là phần khai báo phải nằm trước phần câu lệnh

```
int d=6;
printf("Gia tri d o ngoai: %i\n",d);
{
    int d=74;
    printf("Gia tri d o trong: %i\n",d);
}
printf("Gia tri d o ngoai: %i\n",d);
```



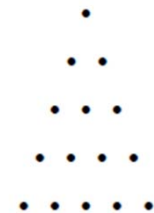
Lệnh lặp

- for
- while
- do

Lệnh lặp

- Sắp xếp 15 điểm để tạo thành hình tam giác ta được hình

- Ta muốn viết chương trình tính và hiển thị số lượng điểm của tam giác thứ 8



```
int triangularNumber;
triangularNumber = 1 + 2 + 3 + 4 + 5 + 6 + 7 + 8;
printf("So diem trong tam giac thu 8 la %i\n",
    triangularNumber);
```



Lệnh lặp for

Lệnh lặp

■ Lệnh lặp for

```
#include <stdio.h>
int main (void)
{
    int n, triangularNumber;
    triangularNumber = 0;

    for ( n = 1; n <= 200; n = n + 1 )
        triangularNumber = triangularNumber + n;

    printf ("So diem trong tam giac thu 200
           la %i\n", triangularNumber);
    return 0;
}
```

Lệnh lặp

- Nếu ta muốn tính số lượng điểm của tam giác thứ 200 ?
- C cho phép ta thực hiện việc này một cách dễ dàng thông qua lệnh lặp
- Trong C có 3 lệnh lặp khác nhau
 - for
 - while
 - Và do

Lệnh lặp

■ Lệnh for

```
#include <stdio.h>
int main (void)
{
    int n, triangularNumber;
    triangularNumber = 0;
    for ( n = 1; n <= 200; n = n + 1 )
        triangularNumber = triangularNumber + n;

    printf (" So diem trong tam giac thu 200 la
           %i\n", triangularNumber);
    return 0;
}
```

Lệnh for

```
for ( biểu_thức_khởi_tạo; điều_kiện_lặp; biểu_thức_lặp )
    lệnh;
```

- **biểu_thức_khởi_tạo**: được sử dụng để đặt giá trị ban đầu khi vòng lặp chưa bắt đầu.
- **điều_kiện_lặp**: là điều kiện cho vòng lặp còn tiếp tục thực hiện.
- **biểu_thức_lặp**: là biểu thức sẽ được thực hiện mỗi khi thực hiện xong một vòng lặp.

Lệnh for

- Quá trình thực hiện lệnh for:

1. Biểu thức khởi tạo được thực hiện đầu tiên
2. Điều kiện lặp sẽ được kiểm tra, nếu không thỏa mãn (giá trị false) thì kết thúc lặp. Ngược lại thì thực hiện lặp
3. Lệnh trong thân vòng lặp được thực hiện
4. Biểu thức lặp được thực hiện
5. Quay trở lại bước 2

Lệnh for

- VD. In ra bảng các tam giác điểm

```
#include <stdio.h>
int main (void)
{
    int n, triangularNumber;
    printf ("BANG SO LUONG DIEM CUA TAM GIAC\n\n");
    printf (" n   Tong tu 1 toi n\n");
    printf ("--- -----\n");
    triangularNumber = 0;
    for ( n = 1; n <= 10; ++n ) {
        triangularNumber += n;
        printf (" %2i %10i\n", n, triangularNumber);
    }
    return 0;
}
```

Lệnh for

- VD. Chương trình in ra kích thước theo yêu cầu người dùng

```
#include <stdio.h>
int main (void)
{
    int n, number, triangularNumber;
    printf ("Nhap vao kích thước tam giác bạn muốn? ");
    scanf ("%i", &number);
    triangularNumber = 0;
    for ( n = 1; n <= number; ++n )
        triangularNumber += n;

    printf ("So lượng điểm của tam giác %i là %i\n", number,
        triangularNumber);

    return 0;
}
```

Lệnh for

■ Lệnh for lồng nhau

```
#include <stdio.h>
int main (void)
{
    int n, number, triangularNumber, counter;
    for ( counter = 1; counter <= 5; ++counter ) {
        printf ("Kich thuc tam giac? ");
        scanf ("%i", &number);
        triangularNumber = 0;
        for ( n = 1; n <= number; ++n )
            triangularNumber += n;
        printf ("So luong diem cua tam giac %i la %i\n\n",
            number, triangularNumber);
    }
    return 0;
}
```

Lệnh for

Một số dạng khác của lệnh **for**

- `for (i=0, j=0; i < 10; ++i)`
`for (i=0, j=100; i < 10; ++i, j=j-10)`
- `for (; j != 100; ++j)`
- `for (int counter=1; counter<= 5; ++counter) *`

Note * : C99 (Dev-C++)



Lệnh while

Lệnh lặp while

■ Lệnh while:

```
while (biểu_thức)
    câu_lệnh;
```

■ Quá trình thực hiện lệnh while:

1. Biểu thức trong ngoặc được thực hiện
 - Nếu kết quả của biểu thức là **TRUE** (hoặc !=0) thì câu lệnh được thực hiện
 - Nếu kết quả là **FALSE** (hoặc ==0) thì kết thúc lệnh lặp
2. Lặp lại bước 1.

Lệnh lặp while

- VD. Chương trình in ra các số từ 1 đến 5

```
#include <stdio.h>
int main (void)
{
    int count = 1;
    while ( count <= 5 ) {
        printf ("%i\n", count);
        ++count;
    }
    return 0;
}
```

Lệnh lặp while

- Tương ứng giữa lệnh lặp for và while

```
for(biểu_thức_khởi_tạo; điều_kiện_lặp; biểu_thức_lặp)
    lệnh;
```

- Lệnh lặp while tương ứng

```
biểu_thức_khởi_tạo;
while(điều_kiện_lặp) {
    lệnh;
    biểu_thức_lặp;
}
```

Lệnh lặp while

- VD. Tìm ước số chung lớn nhất của hai số nguyên dương u và v.

- Thuật toán tìm ước số chung lớn nhất được đề xuất bởi Euclid khoảng năm 300 TCN.

- Bước 1: Nếu v bằng 0 thì thuật toán kết thúc, ước số chung lớn nhất là u
- Bước 2: tính $temp = u \% v$, $u = v$, $v = temp$ và quay trở lại bước 1

Lệnh lặp while

```
#include <stdio.h>
int main (void)
{
    int u, v, temp;
    printf ("Hay nhap vao hai so nguyen duong: ");
    scanf ("%i%i", &u, &v);
    while ( v != 0 ) {
        temp = u % v;
        u = v;
        v = temp;
    }
    printf ("Uoc so chung lon nhat cua chung la %i\n", u);
    return 0;
}
```

Toán tử quan hệ

Toán tử quan hệ : Kiểm tra mối quan hệ giữa hai biến hay giữa một biến và một hằng

Toán tử quan hệ

Toán tử	Ý nghĩa
>	Lớn hơn
>=	Lớn hơn hoặc bằng
<	Nhỏ hơn
<=	Nhỏ hơn hoặc bằng
==	Bằng
!=	Không bằng

Lệnh lặp while

- VD. Viết chương trình in các chữ số của một số nguyên dương theo thứ tự ngược

12346 thì sẽ in ra là 64321



Lệnh do

Lệnh lặp do

```
do
    lệnh;
while ( điều_kiện_lặp );
```

Quá trình thực hiện:

- Bước 1: Lệnh sẽ được thực hiện
- Bước 2: Điều kiện lặp sẽ được kiểm tra
 - Nếu điều kiện lặp là TRUE (hoặc !=0) thì quay trở lại bước 1
 - Ngược lại, điều kiện lặp là FALSE (hoặc bằng 0) thì kết thúc lệnh lặp

Lệnh lặp do

- VD. Chương trình bắt người sử dụng nhập vào số nguyên dương chẵn.

```
#include <stdio.h>
int main (void)
{
    int u;
    do
    {
        printf("Nhap vao so nguyen duong chan: ");
        scanf("%i",&u);
    }
    while(u<0 || u%2!=0);
    printf("So vua nhap: %i\n",u);
    system("pause");
    return 0;
}
```



Lệnh break, continue

Lệnh break, continue

- Trong quá trình lặp, nếu muốn kết thúc vòng lặp sớm (do gặp lỗi, do hết dữ liệu ...) thì ta dùng lệnh **break**

```
int i;
for(i=0; i<100;i++)
{
    printf("%i\n",i);
    if(i==5) break;
}
```

- Trong trường hợp các vòng lặp lồng nhau, chỉ vòng lặp trong cùng nơi mà lệnh break được thực hiện sẽ kết thúc.

Lệnh break, continue

- Lệnh continue làm cho lệnh lặp thực hiện tiếp bỏ qua phần lệnh sau lệnh continue

```
int i;
for(i=0; i<10;i++)
{
    printf("%i\n",i);
    if(i<5) continue;
    printf("bla bla \n");
}
```




Lệnh điều khiển rẽ nhánh

if
if... else
Lệnh if .. else lồng nhau
switch .. case



Lệnh if và if .. else

Lệnh if và if .. else

■ Lệnh if:

```
int main (void)
{
    int number;
    printf ("Type in your number: ");
    scanf ("%i", &number);

    if ( number < 0 )
        number = -number;

    printf ("The absolute value is %i\n", number);
    return 0;
}
```

Lệnh if và if .. else

```
if (biểu_thức_điều_kiện)
    lệnh;
```

■ Quá trình thực hiện:

- Bước 1: biểu_thức_điều_kiện được thực hiện.
- Bước 2:
 - Nếu biểu_thức_điều_kiện có giá trị TRUE (hoặc !=0) thì thực hiện lệnh
 - Ngược lại, thì bỏ qua không làm gì cả

Lệnh if và if .. else

- VD. Chương trình tính tổng điểm trung bình, và đếm số lượng thành viên không qua của lớp.

```
#include <stdio.h>
int main (void)
{
    int soThanhvien, i;
    int slTruot = 0;
    float trungBinh, diem, tongDiem = 0;

    printf ("So luong thanh vien cua lop? ");
    scanf ("%i", &soThanhvien);
```

Lệnh if và if .. else

```
for ( i = 1; i <= soThanhvien; ++i ) {
    printf ("Nhap diem cua thanh vien thu #%i: ", i);
    scanf ("%f", &diem);
    tongDiem = tongDiem + diem;
    if ( diem < 5.0 )
        ++slTruot;
}
trungBinh = tongDiem / soThanhvien;
printf ("\nDiem trung binh = %0.2f\n", trungBinh);
printf ("So luong truot = %i\n", slTruot);
return 0;
}
```

Lệnh if và if .. else

```
if (biểu_thức_điều_kiện)
    lệnh;
else
    lệnh;
```

- Quá trình thực hiện:

- Bước 1: biểu_thức_điều_kiện được thực hiện.
- Bước 2:
 - Nếu biểu_thức_điều_kiện có giá trị TRUE (hoặc !=0) thì thực hiện lệnh sau if
 - Ngược lại, thì thực hiện lệnh sau else

Lệnh if và if .. else

- VD. Chương trình xác định số nguyên nhập vào là chẵn hay lẻ

```
#include <stdio.h>
int main (void)
{
    int soKiemTra, phanDu;
    printf ("Nhap vao so can kiem tra: ");
    scanf ("%i", &soKiemTra);
    phanDu = soKiemTra % 2;

    if ( phanDu == 0 )
        printf ("So vua nhap vao la chan.\n");
    else
        printf ("So vua nhap vao la le.\n");

    return 0;
}
```

Lệnh if và if .. else

Kiểm tra các quan hệ kết hợp (có nhiều hơn 1 quan hệ)

- VD. Kiểm tra các số dương và chẵn

```
if (n>=0 && n%2==0)
    slSo++;
```

- Kiểm tra xem số người dùng có nhập vào có phải là điểm thi không (điểm thi là số trong khoảng 0 - 10)

```
if (diem<0 || diem>10)
    printf("Khong hop le!\n");
else
    <thực hiện tính toán...>
```

Lệnh if và if .. else

- VD. Chương trình kiểm tra năm người dùng nhập vào từ bàn phím có phải năm nhuận hay không (chia hết cho 4 nhưng không chia hết cho 100 hoặc là năm chia hết cho 400).

```
int main (void)
{
    int year;
    printf("Nhap nam can kiem tra: ");
    scanf ("%i", &year);

    if ((year%4 == 0 && year%100 != 0) || year%400 == 0)
        printf ("Nam %i la nam nhuan.\n",year);
    else
        printf ("Khong, day khong phai la nam nhuan.\n");
    return 0;
}
```

Lệnh if và if .. else

Lệnh if lồng nhau

VD1. Đếm số lượng số nguyên dương và chẵn

```
if (n>0)
    if (n%2==0)
        slSo++;
```

VD2. đếm số lượng số chữ số khác 0 của một số

```
slSo=0;
if (n>10) {
    while (n>0) {
        if (n%10 != 0) slSo++;
        n=n/10;
    }
}
else
    if (n>0) slSo=1;
```

Lệnh if và if .. else

- VD3. xác định kết quả học tập theo điểm

```
float diem;
printf("Nhap diem cua sinh vien: ");
scanf("%f", &diem);

if (diem<0 || diem >10)
    printf("Diem khong hop le.\n");
else
    if (diem>=5)
        printf("Sinh vien dat diem do.\n");
    else
        printf("Oh, sinh vien phai hoc lai. ");
```

Lệnh if và if .. else

Lệnh khối else if: viết gọn lại các lệnh if ...else lồng nhau.

```
if ( biểu_thức_điều_kiện_1 )
    lệnh_1;
else if (Biểu_thức_điều_kiện_2 )
    lệnh_2;

<các lệnh else if tiếp theo nếu có>

else
    lệnh_n;
```

Lệnh if và if .. else

■ VD. Xác định các giai đoạn trong đời người

```
if(tuoi<0)
    printf("Tuoi khong hop le.\n");
else if(tuoi<6) //tu 0-6
    printf("Tuoi thieu nhi, nhi dong.\n");
else if(tuoi<12) //tu 6-12
    printf("Tuoi thieu nien.\n");
else if(tuoi<19) //13-19
    printf("Tuoi thanh thieu nien.\n");
else if(tuoi<35) //20-34
    printf("Tuoi thanh nien.\n");
else if(tuoi<60) //35-60
    printf("Tuoi trung nien.\n");
else // >60
    printf("Tuoi cao nien.\n");
```

Lệnh if và if .. else

- VD. Viết chương trình nhập vào từ bàn phím một biểu thức trung tố đơn giản dạng a R b. Tính và in ra giá trị của biểu thức này. Các toán tử R ở đây chỉ là : +, -, *, / và %, các toán hạng a,b là các giá trị số thực

Vd. Biểu thức nhập vào có thể là $34.5 * 7$

Chú ý trường hợp nếu toán tử là phép chia, và toán hạng b có giá trị là 0 (trường hợp chia cho 0)



Lệnh switch

Lệnh switch

```
switch ( biểu_thức )
{
    case giatri_1:
        program statement
        ...
        break;
    case giatri_2:
        ...
        break;
    ...
    case giatri_n:
        program statement ...
        break;
    default:
        program statement ...
        break;
}
```

Lệnh switch

- VD. Chương trình xác định số vừa nhập vào từ bàn phím là số chẵn hay số lẻ.

```
int so;
printf("Nhap vao so nguyen: ");
scanf("%d",&so);
switch (so%2)
{
    case 1:
        printf("So vua nhap la le.\n");
        break;
    case 0:
        printf("So vua nhap la chan.\n");
        break;
}
```

Lệnh switch

- VD. Chương trình tính số ngày trong một tháng của một năm thường (không phải năm nhuận)

```
int thang;
scanf("%d",&thang);
switch (thang)
{
    case 1:
    case 3:
    case 5:
    case 7:
    case 8:
    case 10:
    case 12:
        printf("Thang nay co 31 ngay");
        break;
```

Lệnh switch

```
case 2:
    printf("Thang nay co 28 ngay");
    break;
case 4:
case 6:
case 9:
case 11:
    printf("Thang nay co 30 ngay");
    break;
}
```



Toán tử điều kiện

Toán tử điều kiện

■ Toán tử điều kiện

`điều_kiện ? biểuThức_1 : biểuThức_2;`

VD.

■ Tìm số lớn nhất trong 2 số a,b

`maxValue = (a > b) ? a : b;`

■ Tính căn bậc hai của x nếu x lớn hơn 0

`s = (x < 0) ? -1 : sqrt(x);`

■ Hàm dấu

```
printf("Sign = %i\n", (number < 0) ? -1 :
(number == 0) ? 0 : 1);
```