



Bài giảng

CẤU TRÚC LẬP TRONG NGÔN NGỮ LẬP TRÌNH C

GV: Nguyễn Thị Mỹ Dung

Email: ntmdung@dthu.edu.vn

Học phần: Lập trình căn bản (3 tc)

Sinh viên lớp: ĐHCNTT20



NỘI DUNG MÔN HỌC



01

Một số khái niệm cơ bản (4)

02

Các kiểu dữ liệu và phép toán (4)

03

CÁC LỆNH CẤU TRÚC (6 + 6)

04

Các thao tác trên dữ liệu kiểu mảng (4 + 4)

05

Hàm và phạm vi của biến (3 + 4)



CÁC LỆNH CẤU TRÚC



- I. Cấu trúc rẽ nhánh**
- II. Cấu trúc vòng lặp**
- III. Lệnh goto**
- IV. Bài tập**



Đặt vấn đề



- Bài toán 1: Xét kết quả tích lũy môn học đạt hay không. Nếu điểm trung bình ≥ 4 thì **Đạt**; ngược lại **Không đạt**.

```
#include <stdio.h>
int main() {
    float dtb = 3.9;
    if (dtb <= 10.0 && dtb >= 4.0)
        printf ("Dat\n");
    else
        printf ("Khong dat\n");
    return 0;
}
```

dtb = 3.9;
dtb = 4;
dtb = 7;
dtb = 10;

In ra màn hình



Đặt vấn đề



- Bài toán 2: Hãy trình bày giải thuật tính tổng các số từ 1 đến 5.

```
#include<stdio.h>
#include<conio.h>
void main(){
    int s;
    s = 1;
    s = s + 2;
    s = s + 3;
    s = s + 4;
    s = s + 5;
    printf("\nGia tri s = %d",s);
    getch();
}
```

S = 1;

S = S + 2;

S = S + 3;

S = S + 4;

S = S + 5;

In giá trị S ra
màn hình;



Đặt vấn đề



- Bài toán 3: Hãy trình bày giải thuật tính các số từ 1 đến 1000.

```
#include<stdio.h>
#include<conio.h>
int main(){
    int s;
    s = 1;
    s = s + 2;
    s = s + 3;
    //.....
    s = s + 1000;
    printf("\nGia tri s = %d",s);
    getch();
}
```

S = 1;

S = S + 2;

S = S + 3;

.....

S = S + 1000;

In giá trị S ra
màn hình;



Đặt vấn đề



- Bài toán 3:** Hãy trình bày giải thuật tính tổng các số từ 1 đến 1000.

```
#include<stdio.h>
#include<conio.h>
int main(){
    int s;
    s = 1;
    s = s + 2;
    s = s + 3;
    //.....
    s = s + 1000;
    printf("\nGia tri s = %d",s);
    getch();
}
```

Sử dụng
cấu trúc lệnh nào
để viết chương trình
đơn giản, hiệu quả hơn ?





Cấu trúc rẽ nhánh

- Rẽ nhánh
- Vòng lặp
- Lệnh khác

1. Lệnh if không có else

Cú pháp:

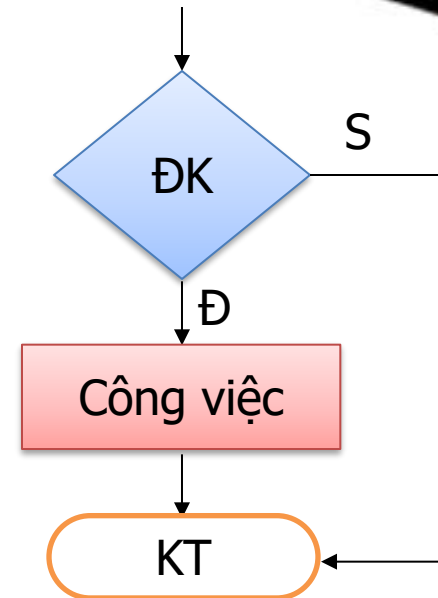
```
if (<điều_kiện>) {  
    <công_việc_1>; //điều kiện đúng  
    <công_việc_2>; //nếu có  
    .... }  
}
```

- Diễn giải:

Nếu điều kiện là **đúng** (TRUE) thì các công việc 1, 2,... sẽ được thi hành; nếu **sai** thì chương trình sẽ bỏ qua.



Khi nào lệnh if có cặp dấu ngoặc { } ?





Cấu trúc rẽ nhánh

- Rẽ nhánh
- Vòng lặp
- Lệnh khác

Ví dụ 1:

Nhập vào 1 số nguyên dương. Xét số nguyên nhập vào là số chẵn hay số lẻ.

```
#include <stdio.h>
int main() {
    int n;
    printf ("\nNhap vao so nguyen duong: ");
    scanf ("%d", &n);
    if (n%2 == 0)
        printf ("\nSo %d la so chan.", n);
    if (n%2 != 0)
        printf ("\nSo %d la so le.", n);
    return 0;
}
```



Cấu trúc rẽ nhánh

- Rẽ nhánh

- Vòng lặp
- Lệnh khác

Ví dụ 2:

Nhập vào 3 số a, b, c. Tìm số lớn nhất trong 3 số.

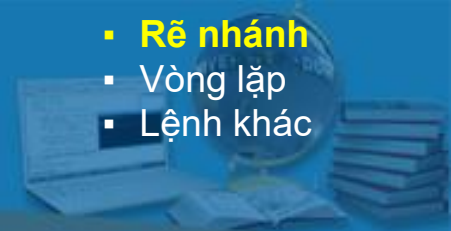
```
#include <stdio.h>
int main() {
    int a, b, c;
    int ln;
    printf ("\nNhap vao 3
so nguyen duong: ");
    scanf ("%d, %d, %d",
&a, &b, &c);
    ln = a;
    if (a > ln)
        ln = a;
```

```
        if (b > ln)
            ln = b;
        if (c > ln)
            ln = c;
    printf ("\nSo lon nhat
trong 3 so (%d, %d,%d)
= %d.", a, b, c,ln);
    return 0;
}
```



Cấu trúc rẽ nhánh

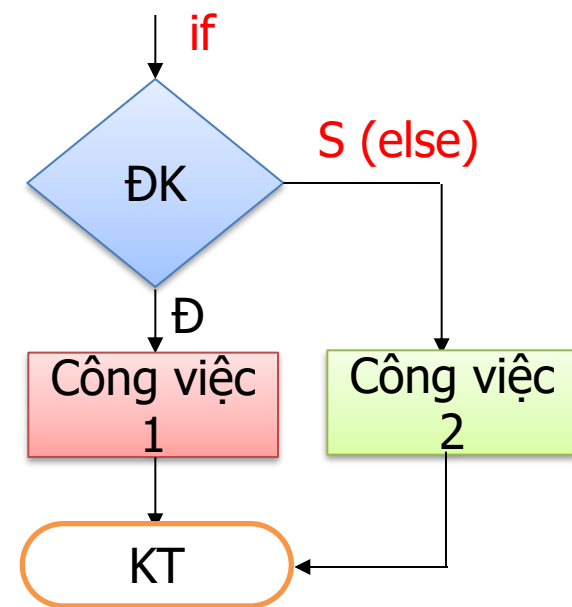
- Rẽ nhánh
- Vòng lặp
- Lệnh khác



2. Lệnh if có else

Cú pháp:

```
if (<điều_kiện>) {  
    <công_việc_1_1>; //đk đúng  
    <công_việc_1_2>; //nếu có  
    ...}  
else {  
    <công_việc_2_1>; //điều kiện sai  
    <công_việc_2_2>; //nếu có  
    ....}
```



- Diễn giải:

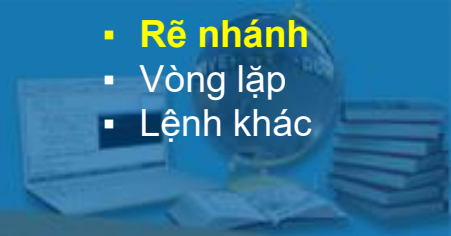
Nếu điều kiện là **đúng** (TRUE) thì nhóm **công việc 1**
ngược lại (điều kiện FALSE) sẽ thi hành nhóm **công việc 2**.



Cấu trúc rẽ nhánh

- Rẽ nhánh

- Vòng lặp
- Lệnh khác



Ví dụ: Viết chương trình nhập vào một số nguyên.
Xét số nguyên đã nhập là số dương hay số âm.

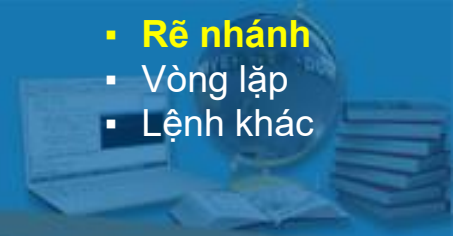
```
#include <stdio.h>
int main() {
    int n;
    printf ("\nNhap vao so nguyen: ");
    scanf ("%d", &n);
    if (n >= 0)
        printf ("\nSo %d la so duong.", n);
    else
        printf ("\nSo %d la so am.", n);
    return 0;
}
```



Cấu trúc rẽ nhánh

- Rẽ nhánh

- Vòng lặp
- Lệnh khác



3. Lệnh if lồng nhau

Cú pháp 1:

```
if (<đk1>
    if (<đk2>
        <công_việc_2_1>;
        ...
    else
        <công_việc_2_2>;
else
    <công_việc_1>;
```

Cú pháp 2:

```
if (<đk1>
    <công_việc_1>;
else if (<đk2>)
    <công_việc_2>;
else
    <công_việc_3>;
...
```



Cấu trúc rẽ nhánh

- Rẽ nhánh

- Vòng lặp
- Lệnh khác

Ví dụ: Giải phương trình bậc nhất: $ax + b = 0$

```
#include <stdio.h>
int main() {
    int a, b;
    float x;
    printf ("\nNhap 2 so a, b: ");
    scanf ("%d, %d", &a, &b);
    if (a == 0)
        if (b == 0)
            printf ("\nPT vo so nghiem.");
        else
            printf ("\nPhuong trinh vo nghiem.");
    else {
        x = (float) -b/a;
        printf ("\nNghiem = %3.2f", x);
    }
    return 0; }
```



Cấu trúc rẽ nhánh

- Rẽ nhánh
- Vòng lặp
- Lệnh khác

➤ Bài tập lệnh if

1.1. Viết chương trình nhập vào tháng, năm. Xét tháng có bao nhiêu ngày.

1.2. Giải và biện luận phương trình bậc 2: $ax^2 + bx + c = 0$.

1.3. Viết chương trình nhập vào ngày, tháng, năm. Xét ngày, tháng, năm có hợp lệ không?

1.4. Viết chương trình nhập vào điểm trung bình. Xếp loại: 10 – 8.5: loại A, 8.4 – 7.0: loại B, 6.9 – 5.5: loại C, 5.4 – 4: loại D, 3.9 – 0: loại F

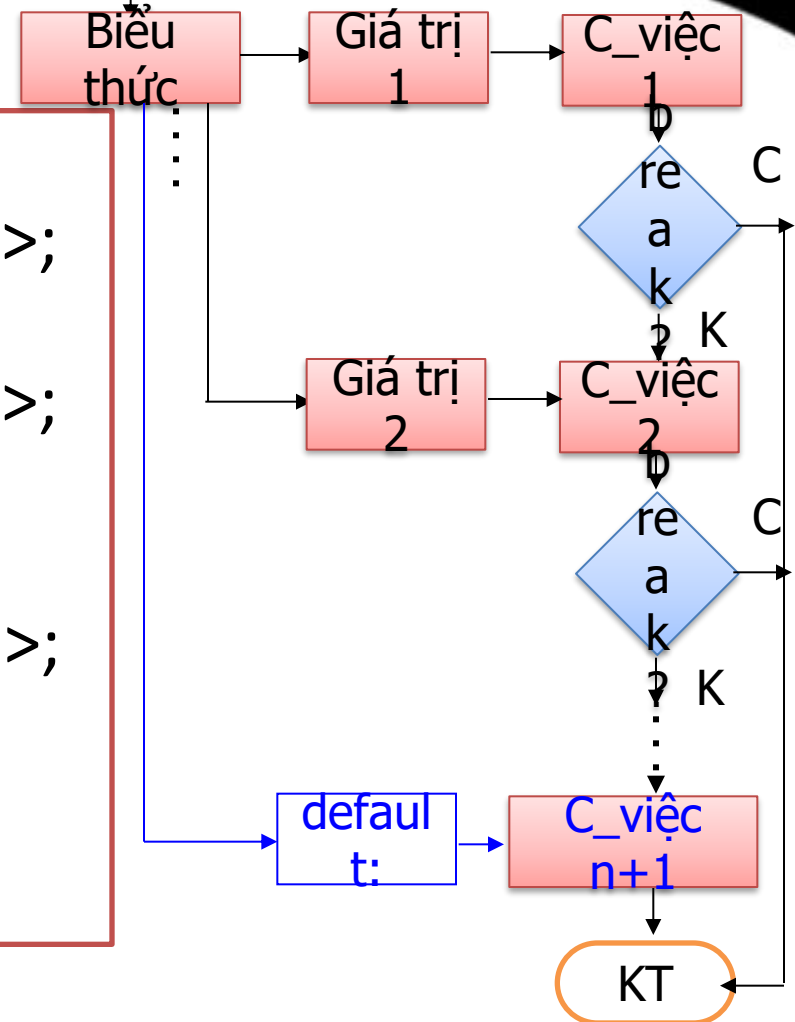


4. Lệnh switch ... case

Cú pháp:

```
switch (<biểu thức>){  
    case <giá_trị_1>: <công_việc_1>;  
        break;  
    case <giá_trị_2>: <công_việc_2>;  
        break;  
    ...  
    case <giá_trị_n>: <công_việc_n>;  
        break;  
    [default: <công_việc_n+1>;]  
}
```

switch...case





Cấu trúc rẽ nhánh

- Rẽ nhánh
- Vòng lặp
- Lệnh khác

Ví dụ: In lên màn hình tháng, năm có bao nhiêu ngày.

```
//thang, nam co bao nhieu ngay
```

```
#include <stdio.h>
```

```
int main() {
```

```
int th, nam;
```

```
printf ("\nNhap thang: ");
```

```
scanf ("%d",&th);
```

```
printf ("\nNhap nam: ");
```

```
scanf ("%d",&nam);
```

```
switch (th) {
```

```
case 1:
```

```
case 3:
```

```
case 5:
```

```
case 7:
```

```
case 8:
```

```
case 10:
```

```
case 12: printf ("\nThang %d co 31 ngay.", th);
```

```
break;
```

```
case 4:
```

```
case 6:
```

```
case 9:
```

```
case 11: printf ("\nThang %d co 30 ngay.", th);  
break;
```

```
case 2:
```

```
if((nam%4==0 && nam%100!=0)  
|| nam % 400 == 0)
```

```
printf ("\nThang %d co 29 ngay.", th);
```

```
else
```

```
printf ("\nThang %d co 28 ngay.", th);
```

```
break;
```

```
default: printf ("\nThang %d khong hop le.", th);
```

```
}
```

```
}
```



Cấu trúc rẽ nhánh

- Rẽ nhánh
- Vòng lặp
- Lệnh khác

➤ Bài tập lệnh lựa chọn

1.5. Viết chương trình nhập vào 2 số a, b và một phép toán. Yêu cầu nếu nhập vào phép toán:

+ : cộng 2 số

– : trừ 2 số

* : nhân 2 số

/ : chia 2 số

1.6. Viết chương trình nhập vào số nguyên. Sử dụng switch...case xét số nguyên nhập vào là số chẵn hay lẻ.

1.7. Viết chương trình nhập 7 số tương ứng hiển thị 7 ngày trong tuần.

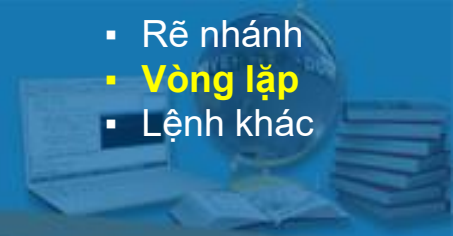
1.8. Tạo menu phép toán cho yêu cầu bài số 1.5.





Cấu trúc vòng lặp

- Rẽ nhánh
- **Vòng lặp**
- Lệnh khác



1. Vòng lặp với số lần lặp xác định trước

1.1. Vòng lặp **for**

1.2. Vòng lặp **for lồng nhau**

2. Vòng lặp với số lần lặp không xác định trước

2.1. Vòng lặp **while**

2.2. Vòng lặp **do ... while**

3. So sánh các vòng lặp (for, while, do...while)



Cấu trúc vòng lặp

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.1. Vòng lặp for

Cú pháp:

```
for ( biểu thức 1 ; biểu thức 2 ; biểu thức 3 ) {  
    câu lệnh, hoặc các câu lệnh;  
}
```

Diễn giải:

biểu thức 1: Khởi tạo giá trị cho biến điều khiển

biểu thức 2: Biểu thức **logic**, điều kiện lặp

biểu thức 3: Biểu thức thay đổi giá trị của biến điều khiển



Khi nào lệnh for có cặp dấu ngoặc { } ?



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.1. Vòng lặp for

Cú pháp:

```
for ( biểu thức 1 ; biểu thức 2 ; biểu thức 3 ) {  
    câu lệnh, hoặc các câu lệnh;  
}
```

• Nguyên lý thực hiện:

- **Bước 1:** Khởi tạo giá trị cho biến điều khiển
- **Bước 2:** Kiểm tra giá trị của biểu thức 2, nếu TRUE thì thực hiện Bước 3, ngược lại thì thoát vòng lặp
- **Bước 3:** Thực hiện Câu lệnh hoặc các câu lệnh.
- **Bước 4:** Thay đổi giá trị của biến điều khiển, quay lại thực hiện Bước 2





1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.1. Vòng lặp for

Lưu đồ

Ví dụ:

```
int i;  
for (i = 1; i < 5; i++)  
    printf("\nLan i = %d",i);
```

```
int i;  
for (i = 1; i < 5; i++) {  
    printf("\nLan");  
    printf("\n%d", i);  
}
```




1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.1. Vòng lặp for

Cú pháp:

```
for ( biểu thức 1 ; biểu thức 2 ; biểu thức 3 ) {  
    câu lệnh, hoặc các câu lệnh;  
}
```

- Ví dụ: Hãy cho biết kết quả thực thi của đoạn chương trình sau

```
int i;  
for (i = 1; i < 5; i++)  
    printf("\nLan i = %d",i);
```

Click xem kết quả



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.1. Vòng lặp for

- Ví dụ: Hãy cho biết kết quả thực thi của đoạn chương trình sau

```
int i;  
for (i = 1; i < 5; i++) {  
    printf("\n Lan");  
    printf("\n %d", i);  
}
```

Click xem kết quả



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.1. Vòng lặp for

- Ví dụ: Hãy cho biết kết quả thực thi của đoạn chương trình sau

```
int i;  
for (i = 1; i < 5; i++)  
    printf("\n Lan i = %d",i);  
printf("\n Ngoai vong lap i = %d", i);
```

Click xem kết quả



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.1. Vòng lặp for

- Ví dụ: Hãy cho biết kết quả thực thi của chương trình sau

```
#include<stdio.h>
#include<conio.h>
int main(){
    int s = 0, i ;
    for (i = 1; i < 10; i++){
        i += 15;
        s += i;
        printf("\n Gia tri i = %d, s = %d", i,
s);
    }
    printf("\nGia tri cua s: %d\n", s);
    getch();
}
```

Click xem kết quả



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.1. Vòng lặp for

- Ví dụ: Hãy viết chương trình (có sử dụng vòng lặp for) tính tổng các số từ 1 đến n , với n là số nguyên dương được nhập từ bàn phím.

Click xem chương trình

Click xem kết quả



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.2. Vòng lặp for không đầy đủ

- Trường hợp 1: Vòng lặp for **thiếu biểu thức 1** (*thiếu biểu thức khởi tạo giá trị của biến điều khiển*).

```
for (      ; biểu thức 2 ; biểu thức 3 ) {  
    câu lệnh, hoặc các câu lệnh;  
}
```



Làm thế nào để khởi tạo giá trị của biến điều khiển ?



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.2. Vòng lặp for không đầy đủ

- Trường hợp 1: Thiếu biểu thức khởi tạo giá trị của biến điều khiển

```
for (      ; biểu thức 2 ; biểu thức 3 ) {  
    câu lệnh, hoặc các câu lệnh;  
}
```

Ví dụ: Hãy sử dụng vòng lặp for *thiếu biểu thức khởi tạo giá trị của biến điều khiển*, viết chương trình tính **tổng các số nguyên từ 1 đến n** (với *n* là số nguyên dương được nhập từ bàn phím).



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.2. Vòng lặp for không đầy đủ

Ví dụ: Hãy sử dụng vòng lặp for thiếu biểu thức khởi tạo giá trị của biến điều khiển, viết chương trình tính **tổng các số nguyên từ 1 đến n** (với n là số nguyên dương được nhập từ bàn phím).

Click xem chương trình



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.2. Vòng lặp for không đầy đủ

- Trường hợp 2: Vòng lặp for **thiếu biểu thức 1** và **thiếu biểu thức 3** (*thiếu biểu thức khởi tạo giá trị của biến điều khiển, thiếu biểu thức thay đổi giá trị của biến điều khiển*).

```
for (      ; biểu thức 2 ;      ) {  
    câu lệnh, hoặc các câu lệnh;  
}
```



Sử dụng vòng lặp for này như thế nào?



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.2. Vòng lặp for không đầy đủ

- Trường hợp 2:

```
for (      ; biểu thức 2 ;      ) {  
    câu lệnh, hoặc các câu lệnh;  
}
```

Ví dụ: Hãy sử dụng vòng lặp for thiếu biểu thức khởi tạo giá trị của biến điều khiển và thiếu biểu thức thay đổi (tăng/giảm) giá trị của biến điều khiển, viết chương trình tính **tổng các số nguyên từ 1 đến n** (với *n* là số nguyên dương được nhập từ bàn phím).



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.2. Vòng lặp for không đầy đủ

Ví dụ: Hãy sử dụng vòng lặp for thiếu biểu thức khởi tạo giá trị của biến điều khiển và thiếu biểu thức thay đổi (tăng/giảm) giá trị của biến điều khiển, viết chương trình tính **tổng các số nguyên từ 1 đến n** (với *n* là số nguyên dương được nhập từ bàn phím).

Click xem chương trình



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.2. Vòng lặp for không đầy đủ

- Trường hợp 3:

```
for (      ;      ;      ) {  
    câu lệnh, hoặc các câu lệnh;  
}
```



Có thể sử dụng vòng lặp với cú pháp như mẫu trên hay không ?

Nếu được thì sử dụng như thế nào?



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.2. Vòng lặp for không đầy đủ

- Trường hợp 3:

```
for (      ;      ;      ) {  
    câu lệnh, hoặc các câu lệnh;  
}
```

Ví dụ: Hãy sử dụng vòng lặp for thiếu cả 3 biểu thức (với cú pháp như trên), viết chương trình tính **tổng các số nguyên từ 1 đến n** (với *n* là số nguyên dương được nhập từ bàn phím).



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.2. Vòng lặp for không đầy đủ

```
for (      ;      ;      ) {  
    câu lệnh, hoặc các câu lệnh;  
}
```

Ví dụ: Hãy sử dụng vòng lặp for *thiếu cả ba biểu thức*, viết chương trình tính **tổng các số nguyên từ 1 đến n** (với *n* là số nguyên dương được nhập từ bàn phím).

Click xem chương trình



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.3. Vòng lặp for với mỗi biểu thức có nhiều lệnh

- Ví dụ: Hãy cho biết kết quả của chương trình sau

```
#include<stdio.h>
#include<conio.h>
int main(){
    int a, b;
    for (a = 1, b = 9; a<=b; a++, b--)
        printf("\n a = %d, b = %d", a, b);
    getch();
}
```

Click xem kết quả



1. Vòng lặp với số lần lặp xác định trước

- **Vòng lặp for**
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

1.4. Vòng lặp for lồng nhau

- Ví dụ: Hãy cho biết kết quả của chương trình sau

```
#include<stdio.h>
#include<conio.h>
void main(){
    int s = 0, i, j;
    int n = 3, m = 3;
    for (i = 1; i<=n; i++){
        for (j = 1; j <= m; j++)
            s += i;
    }
    printf("\n Gia tri s = %d", s);
    getch();
}
```

Click xem kết quả



Bài tập vòng lặp for



1/ In tam giác sao theo dạng (giả sử $n = 4$)

```
      *
     *  *
    *    *
   *      *
  * * * * *
```

1
2
3
4

1							7	1
1	2					6	7	2
1	2	3		5	6	7		3
1	2	3	4	5	6	7		4
1	2	3		5	6	7		5
1	2				6	7		6
1						7		7

2/ Tính tổng các số chẵn từ 1 \rightarrow n, với n nhập vào.

3/ Tính tổng các số lẻ từ 1 \rightarrow n, với n nhập vào.

4/ Tính tổng dãy số theo dạng:

$$S = 1 - 2 + 3 - \dots + 2n-1, \text{ với } n \text{ nhập vào.}$$



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- **Vòng lặp while**
- Vòng lặp do...while
- So sánh vòng lặp

2.1. Vòng lặp while

Cú pháp:

```
while (biểu thức logic) {  
    câu lệnh, hoặc các câu lệnh;  
}
```



Khi nào lệnh while có cặp dấu ngoặc { } ?



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

2.1. Vòng lặp while

- Cú pháp:

```
while (biểu thức logic) {  
    câu lệnh, hoặc các câu lệnh;  
}
```

- Nguyên lý thực hiện:

- **Bước 1:** Kiểm tra giá trị của biểu thức logic, **nếu** giá trị là **TRUE** (giá trị khác 0) thì thực hiện bước 2, **ngược lại** thoát vòng lặp.
- **Bước 2:** Thực hiện câu lệnh hoặc các câu lệnh bên trong vòng lặp.

Lưu ý: nếu câu lệnh là lệnh thoát vòng lặp thì thoát vòng lặp, ngược lại thì thực hiện Bước 1.



Hãy vẽ lưu đồ nguyên lý thực hiện của vòng lặp while ?



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- **Vòng lặp while**
- Vòng lặp do...while
- So sánh vòng lặp

2.1. Vòng lặp while

Lưu đồ

Ví dụ:

```
int a = 10;  
while (a < 20 ) {  
    printf("Gia tri cua a: %d\n", a);  
    a++;  
}
```



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- **Vòng lặp while**
- Vòng lặp do...while
- So sánh vòng lặp

2.1. Vòng lặp while

Ví dụ: Hãy cho biết kết quả của chương trình sau

```
#include<stdio.h>
#include<conio.h>
int main(){
    int i = 1;
    while (i < 5){
        printf("\n Gia tri i = %d",i);
        i++;
    }
    getch();
}
```

Click xem kết quả



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- **Vòng lặp while**
- Vòng lặp do...while
- So sánh vòng lặp

2.1. Vòng lặp while

- Ví dụ: Hãy cho biết kết quả của chương trình sau

```
#include<stdio.h>
#include<conio.h>
int main(){
    int i = 1, kt = 2;
    while (kt) {
        printf("\n Gia tri i = %d",i);
        i++;
        if (i > 3)
            kt = 0; //điều kiện thoát vòng
    }
    printf("\n Ngoai vong lap i = %d",i);
    getch();
}
```

Click xem kết quả



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- **Vòng lặp while**
- Vòng lặp do...while
- So sánh vòng lặp

2.1. Vòng lặp while

- Ví dụ: Hãy cho biết kết quả của chương trình sau

```
#include<stdio.h>
#include<conio.h>
int main(){
    int i = 10;
    while (i > 1){
        i-- ;
        printf("\n Gia tri i = %d",i);
        if (i < 8)
            break;
    }
    printf("\n Ngoai vong lap i = %d\n",i);
    getch();
}
```

Click xem kết quả



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- **Vòng lặp while**
- Vòng lặp do...while
- So sánh vòng lặp

2.1. Vòng lặp while

- Ví dụ: Hãy cho biết kết quả của chương trình sau

```
#include<stdio.h>
#include<conio.h>
int main(){
    int i = 10;
    while (i > 1){
        i-- ;
        printf("\n Gia tri i = %d\n",i);
        if (i < 8)
            return;
    }
    printf("\n Ngoai vong lap i = %d\n",i);
    getch();
}
```

Click xem kết quả



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

2.2. Vòng lặp do ... while

Cú pháp:

```
do {  
    câu lệnh, hoặc các câu lệnh;  
} while (biểu thức logic);
```

- Nguyên lý thực hiện:

- **Bước 1:** Thực hiện câu lệnh hoặc các câu lệnh sau **do**.
- **Bước 2:** Kiểm tra giá trị của biểu thức logic, **nếu** giá trị là **TRUE** (giá trị khác 0) thì thực hiện bước 2, **ngược lại** thoát vòng lặp.

Lưu ý: Vòng lặp do ... while thực hiện công việc ít nhất 1 lần.



Hãy vẽ lưu đồ nguyên lý thực hiện của vòng lặp do ... while ?



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- Vòng lặp while
- **Vòng lặp do...while**
- So sánh vòng lặp

2.1. Vòng lặp do ... while

Lưu đồ

Ví dụ:

```
int so;  
do{  
    printf ("\nNhap so=") ;  
    scanf ("%d", &so) ;  
}while (so< 1);
```



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- Vòng lặp while
- **Vòng lặp do...while**
- So sánh vòng lặp

2.2. Vòng lặp do ... while

Ví dụ: Hãy cho biết kết quả của chương trình sau:

```
#include <stdio.h>
int main(){
    int so;
    do{
        printf("\nNhap so = ");
        scanf("%d", &so);
    }while(so < 1);
    printf("\nSo nhap = %d", so);
}
```

Click xem kết quả



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- Vòng lặp while
- **Vòng lặp do...while**
- So sánh vòng lặp

2.2. Vòng lặp do ... while

Ví dụ: Hãy cho biết kết quả của chương trình sau:

```
#include <stdio.h>
int main(){
    float so, tong= 0.0;
    do {
        printf ("Nhap so thuc: ");
        scanf ("%f",&so);
        tong += so;
    }while(so != 0.0);
    printf ("Tong cac so = %3.2f",tong);
    return 0;
}
```

Click xem kết quả



2. Vòng lặp với số lần lặp không xác định trước

- Vòng lặp for
- Vòng lặp while
- Vòng lặp do...while
- So sánh vòng lặp

2.3. So sánh các vòng lặp

for, while, do ... while
giống nhau ? Khác nhau?

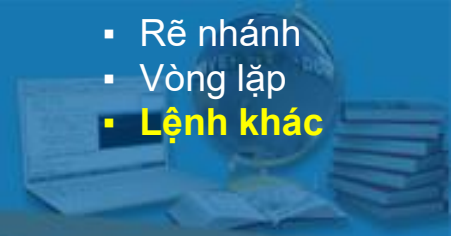


Hãy viết đoạn chương trình tính tổng từ 1 đến 5
bằng cách sử dụng các vòng lặp???



Nhóm lệnh khác

- Rẽ nhánh
- Vòng lặp
- **Lệnh khác**



1.1. Thoát vòng lặp

Hãy cho biết dùng lệnh gì để
thoát vòng lặp ?



Click xem đáp án



Nhóm lệnh khác

- Rẽ nhánh
- Vòng lặp
- **Lệnh khác**



1.1. Thoát vòng lặp (break)

- Ví dụ: Hãy cho biết kết quả thực thi của chương trình sau

```
#include<stdio.h>
#include<conio.h>
int main(){
    int i;
    for (i = 10; i >= 1; i--){
        printf("\n Lan %d",i);
        if (i < 8)
            break;
    }
    getch();
}
```

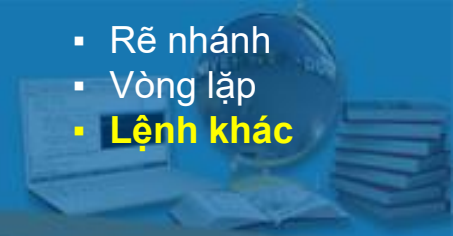
Dùng để chấm dứt thi hành cụm lệnh nào đó trong lệnh điều kiện **switch hoặc vòng lặp **for, while,...****

Click xem kết quả



Nhóm lệnh khác

- Rẽ nhánh
- Vòng lặp
- **Lệnh khác**



1.1. Thoát vòng lặp (break)

Ví dụ khác: Hãy cho biết kết quả thực thi của chương trình sau

```
#include<stdio.h>
#include<conio.h>
int main(){
    int i;
    for (i = 10; i >= 1; i--){
        if (i < 8)
            break;
        printf("\n Lan %d",i);
    }
    getch();
}
```

Click xem kết quả



Nhóm lệnh khác

- Rẽ nhánh
- Vòng lặp
- **Lệnh khác**

1.2. Lệnh **continue**

- Ví dụ: Hãy cho biết kết quả thực thi của chương trình sau

```
#include<stdio.h>
#include<conio.h>
int main(){
    int i;
    for (i = 1; i < 5; i++){
        if (i < 3)
            continue;
        printf("\n Lan %d",i);
    }
    getch();
}
```

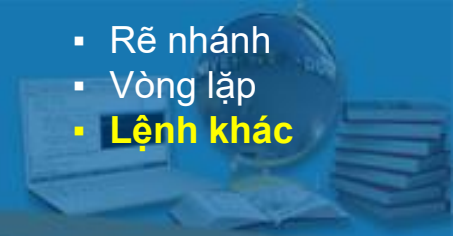
Thay vì chấm dứt
một vòng lặp giữa
chừng, có khi ta cần
tiếp tục vòng lặp
mà **nhảy qua một**
số câu lệnh nào đó.

Click xem kết quả



Nhóm lệnh khác

- Rẽ nhánh
- Vòng lặp
- **Lệnh khác**



1.2. Lệnh **continue**

Ví dụ: Hãy cho biết kết quả thực thi của chương trình sau

```
#include<stdio.h>
#include<conio.h>
int main(){
    int s = 0; int i;
    for (i = 1; i < 10; i++){
        if (i < 5)
            continue;
        else
            s += i;
    }
    printf("\nGia tri cua s: %d\n", s);
    getch();
}
```

Click xem kết quả



Nhóm lệnh khác

- Rẽ nhánh
- Vòng lặp
- **Lệnh khác**



1.3. Lệnh **return**

- Cú pháp và Ví dụ:

return;

return <biểu_thức>;

return (<biểu_thức>;)

VD:

return;

return 0;

return (a+b);

Lệnh return dùng để thoát ra khỏi 1 hàm để trở về hàm gọi nó.



Nhóm lệnh khác

- Rẽ nhánh
- Vòng lặp
- **Lệnh khác**

1.4. Lệnh **goto**

Lệnh goto làm cho chương trình nhảy đến vị trí xác định bởi tên_nhãn. Cú pháp: **goto <label>;**

1.5. Lệnh **ỗng**

- Là lệnh chỉ gồm có dấu chấm phẩy (;).
- Lệnh này thường được dùng trong thân vòng lặp mà ở đó không muốn có một lệnh nào.

Click xem ví dụ



Bài tập cuối chương



3.9. Viết chương trình tính giai thừa của một số nguyên dương.

3.10. Viết ch/tr tính tổng 2 số theo nguyên tắc sau: $1 + 9$, $2 + 8$, $3 + 7$, $4 + 6$, $5 + 5$, khi 2 số bằng nhau thì dừng.

3.11. Viết ch/tr in bảng mã ASCII từ 32 □ 255

3.12. Viết chương trình nhập vào số n khác 0 và thực hiện tính: tổng các số dương đã nhập, đếm số lượng số dương, tính trung bình các số dương cho đến $n = 0$ thì dừng và in ra kết quả mỗi yêu cầu 1 dòng.



Bài tập cuối chương



3.13. Viết chương trình in ra bảng cửu chương từ 2□ 9 như sau:

$$2 \times 1 = 2$$

$$2 \times 2 = 4$$

...

$$2 \times 9 = 18$$

$$3 \times 1 = 3$$

$$3 \times 2 = 6$$

...

$$3 \times 9 = 27$$

...

...

...

$$9 \times 1 = 9$$

$$9 \times 2 = 18$$

...

$$9 \times 9 = 81$$

3.14. Viết ch/tr in tam giác sao với chiều cao nhập.

VD: h = 4

a) *
* *
* * *
* * * *

b) * * * *
* * *
* *
*

c) *
* *
* *
* * * *

d) * * * *
* *
* *
*

3.15. Viết chương trình in giá trị dãy Fibonanci với $F_1 = F_2 = 1$; $F_n = F(n-1) + F(n-2)$.

VD: n = 5 => Dãy Fibo: 1 1 2 3 5



Bài tập cuối chương



3.16. Viết chương trình nhập 1 ký tự. Xét ký tự thuộc tập: chữ cái thường, chữ cái hoa, chữ số, tập khác.

3.17. Viết ch/tr nhập vào 1 số, xét số nhập có là số nguyên tố.

3.18. Viết ch/tr nhập vào một số, xét số nhập có là số chính phương.

3.19. Viết ch/tr nhập vào một số, xét số nhập có là số hoàn hảo.

3.20. Viết ch/tr nhập vào 1 số có nhiều chữ số. Yêu cầu:

- Đếm có bao nhiêu chữ số
- Tổng các chữ số, trung bình các chữ số.

3.21. Viết ch/tr nhập vào 2 số. Tìm USCLN của 2 số.



Tổng kết chương



- Tìm hiểu lệnh lựa chọn: **if, if ... else, switch**
- Tìm hiểu cấu trúc vòng lặp: **for, while, do ... while**
- **Phân biệt được các cấu trúc vòng lặp và vận dụng vào giải quyết bài toán thích hợp**
- Tìm hiểu các lệnh hỗ trợ: **break, continue, goto, return, lệnh rỗng.**

