VÒNG LẶP

# Lý thuyết:

Vòng lặp là một đoạn mã lệnh trong chương trình được thực hiện lặp đi lặp lại cho đến khi thỏa mãn một điều kiện nào đó. Vòng lặp là một khái niệm cơ bản trong lập trình cấu trúc.



Trong C có các loại vòng lặp sau:

Vòng lặp **for**

Vòng lặp **while**

Vòng lặp **do…while**

1. Vòng lặp **for**

Một vòng lặp **for** trong C là một cấu trúc điều khiển lặp đi lặp lại mà cho phép bạn viết một vòng lặp một cách hiệu quả, mà cần thực hiện trong một khoảng thời gian cụ thể nào đó.

## Cú pháp:

Cú pháp của một vòng lặp **for** trong Ngôn ngữ C là:

for ( khoi\_tao\_bien; dieu\_kien; tang/giam )

{

cac\_lenh;

}

int main ()

{

/\* phan thuc thi vong lap for \*/

for( int a = 4; a < 16; a = a + 1 )

{

printf("Gia tri cua a la: %d\n", a);

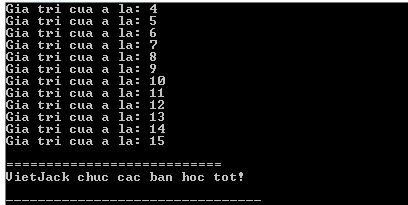
}

printf("\n===========================\n");

return 0;

}

Biên dịch và thực thi chương trình C trên sẽ cho kết quả sau:



2.Vòng lặp **while**

Một lệnh vòng lặp **while** trong Ngôn ngữ chương trình C thực hiện lặp đi lặp lại một lệnh mục tiêu đến khi nào điều kiện đã cho còn là đúng.

## Cú pháp:

Cú pháp của một vòng lặp **while** trong Ngôn ngữ C là:

while(dieu\_kien)

{

cac\_lenh;

}

int main ()

{

/\* phan dinh nghia bien cuc bo \*/

int a = 5;

/\* phan thuc thi vong lap while \*/

while( a < 16 )

{

printf("Gia tri cua a la: %d\n", a);

a++;

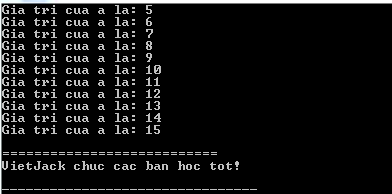
}

printf("\n===========================\n");

return 0;

}

Biên dịch và thực thi chương trình C trên sẽ cho kết quả sau:



3. Vòng lặp **do…while**

Không giống như các vòng lặp **for** và **while**, mà kiểm tra điều kiện vòng lặp ở ngay bước đầu tiên của vòng lặp, vòng lặp **do…while** trong Ngôn ngữ C kiểm tra điều kiện của nó tại phần cuối của vòng lặp.

Một vòng lặp **do...while** là tương tự như vòng lặp while, ngoại trừ ở điểm một vòng lặp do…while bảo đảm thực hiện vòng lặp ít nhất một lần.

## Cú pháp:

Cú pháp của một vòng lặp **do...while** trong Ngôn ngữ C là như sau:

do

{

cac\_lenh;

}while( dieu\_kien );

Bạn chú ý rằng, biểu thức điều kiện xuất hiện ở cuối cùng của vòng lặp, vì thế các lệnh trong vòng lặp thực hiện một lần trước khi điều kiện được kiểm tra.

int main ()

{

/\* phan dinh nghia bien cuc bo \*/

int a = 10;

/\* vong lap do...while \*/

do

{

printf("Gia tri cua a la: %d\n", a);

a = a + 1;

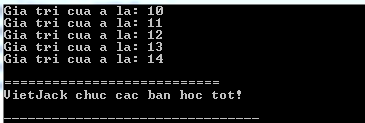
}while( a < 15 );

printf("\n===========================\n");

return 0;

}

Biên dịch và thực thi chương trình C trên sẽ cho kết quả sau:



|  |  |
| --- | --- |
| **Lệnh điều khiển** | **Miêu tả** |
| [**Lệnh break**](http://vietjack.com/lap_trinh_c/lenh_break_trong_c.jsp) | Kết thúc **vòng lặp** hoặc lệnh **switch** và chuyển sang thực thi vòng lặp hoặc lệnh switch ngay sau nó. |
| [**Lệnh continue**](http://vietjack.com/lap_trinh_c/lenh_continue_trong_c.jsp) | Khi gặp lệnh này thì chương trình sẽ bỏ qua các câu lệnh ở dưới nó (trong cùng một câu lệnh lặp) để thực hiện vòng lặp mới. |
| [**Lệnh goto**](http://vietjack.com/lap_trinh_c/lenh_goto_trong_c.jsp) | Chuyển tới lệnh được gán. Mặc dù vậy, nó được khuyên rằng không nên sử dụng lệnh goto trong chương trình của bạn. |

Ví dụ:

**Lệnh break :**

int main ()

{

/\* phan dinh nghia bien cuc bo \*/

int a = 5;

/\* vong lap while \*/

while( a < 19 )

{

printf("Gia tri cua a la: %d\n", a);

a++;

if( a > 14)

{

/\* ket thuc trinh thuc thi cua vong lap voi lenh break \*/

break;

}

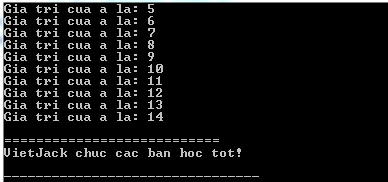
}

printf("\n===========================\n");

return 0;

}

Biên dịch và thực thi chương trình C trên sẽ cho kết quả sau:



**Lệnh Continue:**

int main ()

{

/\* phan dinh nghia bien cuc bo \*/

int a = 5;

/\* vong lap do...while \*/

do

{

if( a == 10)

{

/\* nhay qua buoc lap nay \*/

printf("Nhay qua buoc lap nay!\n");

a = a + 1;

continue;

}

printf("Gia tri cua a la: %d\n", a);

a++;

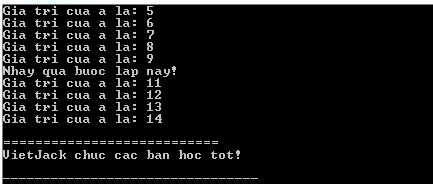
}while( a < 15 );

printf("\n===========================\n");

return 0;

}

Biên dịch và thực thi chương trình C trên sẽ cho kết quả sau:



**Lệnh Goto**

Cú pháp cho một lệnh **goto** trong Ngôn ngữ C như sau:

goto ten\_nhan;

..

.

ten\_nhan: lenh;

int main ()

{

/\* phan dinh nghia bien cuc bo \*/

int a = 5;

/\* vong lap do...while \*/

Label1:do

{

if( a == 10)

{

/\* nhay qua buoc lap nay \*/

printf("Nhay qua buoc lap nay!\n");

a = a + 1;

goto Label1;

}

printf("Gia tri cua a la: %d\n", a);

a++;

}while( a < 15 );

printf("\n===========================\n");

return 0;

}

Biên dịch và thực thi chương trình C trên sẽ cho kết quả sau:



Cú pháp để **lồng vòng lặp for** trong C như sau:

for ( khoi\_tao\_bien; dieu\_kien; tang/giam )

{

for ( khoi\_tao\_bien; dieu\_kien; tang/giam )

{

cac\_lenh;

}

cac\_lenh;

}

Cú pháp để **lồng vòng lặp while** trong C như sau:

while(dieu\_kien)

{

while(dieu\_kien)

{

cac\_lenh;

}

cac\_lenh;

}

Cú pháp để lồng vòng lặp **do…while** trong C như sau:

do

{

cac\_lenh;

do

{

cac\_lenh;

}while( dieu\_kien );

}while( dieu\_kien );

# Bài tập thực hành:

**BÀI TẬP 1**

**1. Thời lượng:** 15 phút

**2. Mô tả bài toán:**

Viết chương trình in ra theo thứ tự bảng các ký tự thường (a-z) và ký tự hoa (A-Z ) .

**3. Gợi ý:**

Xem trong bảng mã ASCII -> để quyết định dòng for

printf (“%c ” , i ) //in giá trị i theo %c sẽ ra được ký tự

**BÀI TẬP 2**

**1.Thời lượng :** 15 phút

**2.Mô tả bài toán :** Nhập vào 1 số n. Tính giai thừa của nó. (1\*2\*3\*4\*..\*n)

**3.Gợi ý :**

- Sử dụng vòng lặp với i = 1 chạy đến n

- Mỗi bước lặp thực hiện : S = S \* i (ban đầu khai báo S = 0)

**BÀI TẬP 3**

**1. Thời lượng:** 15 phút

**2. Mô tả bài toán:** Tính tổng các số chia hết cho 4 và không chia hết cho 5 nhỏ hơn n

**3. Gợi ý:**

* Chạy vòng lặp i =1 đến n
* Nếu i chia hết cho 4 không chia hết cho 5, thì Tổng = Tổng + i
* Vòng lặp dừng , output : Xuất tổng

**BÀI TẬP 4**

**1.Thời lượng :** 15 phút

**2.Mô tả bài toán :**

1. Nhập số nguyên N>=0, tính S2 = 2 + 4 + 6 + ... + N
2. Nhập số nguyên N>=0, tính S3 = 1 + 3 + 5 + ... + N
3. Nhập số nguyên N>=0, tính S4 = 12 + 22 + 32 + ... + N2

**3.Gợi ý :**

Sử dụng vòng lặp giải quyết bài toán :

Ví dụ : for ( i =… ; i < … ; i = i + 2 )

**BÀI TẬP 5**

**1. Thời lượng:** 15 phút

**2. Mô tả bài toán:**

1. Nhập số nguyên N>=0, tính S5 = 1 + 1/2 + 1/3 + ... +1/ N
2. Nhập số nguyên N>=0, tính S6 = 1/2 + 1/4 + 1/6 + ... + 1 / 2N
3. Nhập số nguyên N>=0, tính S7 = 1 + 1/3 + 1/5 + ... + 1 / 2N + 1

**3. Gợi ý:**

Sử dụng vòng lặp giải quyết bài toán :

Ví dụ : for ( i =… ; i < … ; i = i + 2 )

S = S + 1/(float) i

**BÀI TẬP 6**

**1.Thời lượng :** 15 phút

**2.Mô tả bài toán :** Nhập số nguyên n>0, tính tổng các ước số của n.

**3.Gợi ý :**

for ( i =1 ; i < n ; i ++ )

Nếu n chia hết cho i => i chính là ước số của n => tổng các giá trị ước số

**BÀI TẬP 7**

**1. Thời lượng:** 15 phút

**2. Mô tả bài toán:**

Nhập số nguyên N>0, cho biết N có phải là số hoàn chỉnh hay không ? Biết số hoàn chỉnh là số có tổng các ước số nhỏ hơn nó = với chính nó

Ví dụ: 6 có ước ( 1,2,3,**~~6~~** ) 1+2+3 = 6 -> 6 là số hoàn chỉnh.

**3. Gợi ý:**

- Nhập n

- Tính tổng các ước số < n . Nếu tổng ước bằng với n => kết luận là số hoàn chỉnh

**BÀI TẬP 8**

**1.Thời lượng :** 15 phút

**2.Mô tả bài toán :** In ra dãy số giảm dần :100, 95 , 90, 85,………., 5.

**3.Gợi ý :**

* Dùng for ngược i = 100 về 5
* Mỗi lần lặp : xuất i => giảm i = i – 5

**BÀI TẬP 9**

**1. Thời lượng:** 15 phút

**2. Mô tả bài toán:** Cho trước mẫu password là 9. Nhập vào 1 password . nếu pass đúng thì báo “Login thành công”, ngược lại: nhập sai 3 lần báo “Khoá tài khoản”.

**4. Gợi ý:**

- Khai báo biến password mẫu = 9

- Yêu cầu người dùng nhập password

- Sử dụng if so sánh => Nếu password nhập không đúng với pass mẫu => sử dụng vòng lặp để yêu cầu nhập lại password . Đồng thời đếm số lần người dùng đã nhập password => Nếu số lần đếm =3 thì thoát khỏi vòng lặp , và báo “khóa tài khoản”.

- Nếu password nhập đúng với password mẫu => thoát vòng lặp => thông báo “log in thành công”

**BÀI TẬP 10**

**1.Thời lượng :** 15 phút

**2.Mô tả bài toán :** Nhập vào 2 số . Hãy tìm tổng các số lẻ giữa 2 số đó.

(Ví dụ : nhập vào 10 và 200 . Tìm tổng các số lẻ giữa 2 số 10…200 )

**3.Gợi ý :**

- Nhập vào 2 giá trị startvà end

\* Điều kiện start < end :

- Sử dụng vòng lặp i = start đến end

- Với mỗi lần lặp : kiểm tra i là số lẻ => tổng các i lẻ

**BÀI TẬP 11**

**1. Thời lượng:** 15 phút

**2. Mô tả bài toán:** In bảng cửu chương ( theo số người dùng nhập vào )

Ví dụ: Nhập 2 -> in bảng cử chương 2

**3. Gợi ý:**

- Sử dụng vòng lặp for i=1 => đến 10

**BÀI TẬP 12**

**1.Thời lượng :** 15 phút

**2.Mô tả bài toán :** Nhập vào số n (với n>0).Đếm các chữ số của số n

**3.Gợi ý :**

- Sử dụng vòng lặp while

- Mỗi lần lặp : n = n/10 => thực hiện đếm => đến khi n = 0 thoát vòng lặp => xuất biến đếm

**BÀI TẬP 13**

**1. Thời lượng:** 15 phút

**2. Mô tả bài toán:** Kiểm tra số nguyên dương n có toàn chữ số chẵn hay ko?

**3. Gợi ý:**

- Sử dụng vòng lặp while

- Mỗi lần lặp :

- number = n MOD 10 => nếu number là số lẻ => thoát vòng lặp=> báo n không toàn số chẵn

- Vòng lặp sẽ lặp đến khi n = 0 , thì thông báo n toàn chữ số chẵn.

**BÀI TẬP 14**

**1.Thời lượng :** 15 phút

**2.Mô tả bài toán :**

1. Viết chương trình vẽ một hình chữ nhật bằng các dấu \* với chiều dài và chiều rộng do người dùng nhập vào

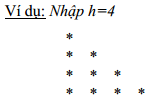
\* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \*

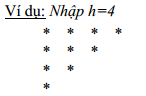
\* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \*

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



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



1. Viết chương trình in ra màn hình:



1. Viết chương trình in ra màn hình hình chữ nhật rỗng kích thước n × m (m, n nhập từ bàn phím).

Ví dụ: Nhập m=5, n=4



**3.Gợi ý :**

Sử dụng 2 vòng lặp for lồng nhau :

for ( i …; ….; ….)

{

for ( j …; ….; ….)

}

**Link tham khảo**:

[**http://vietjack.com/lap\_trinh\_c/vong\_lap\_trong\_c.jsp**](http://vietjack.com/lap_trinh_c/vong_lap_trong_c.jsp)