

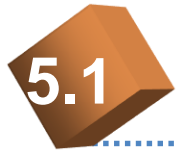
# Phần 2. NN lập trình C++

## CHƯƠNG 5 CÂU LỆNH LẶP





# Chương 5 – Câu lệnh lặp



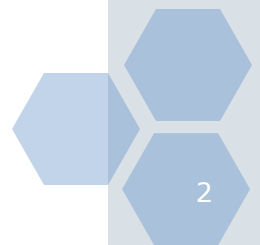
## Lệnh for



## Lệnh while



## Lệnh do ... while





# Chương 5 – Câu lệnh lặp

## ❖ Câu lệnh lặp – loop statement

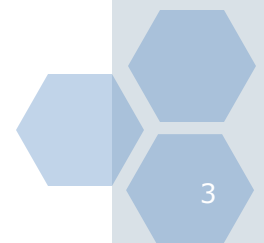
- Hành động được lặp đi lặp lại trong khi một điều kiện nào đó còn được thỏa mãn

## ❖ Ví dụ:

- In danh sách các số từ 1 đến 10 ra màn hình, mỗi số trên một dòng
- Nếu danh sách vẫn còn số thì tiếp tục in ra cho đến hết danh sách.

```
cout<<1<<endl;  
cout<<2<<endl;  
...  
cout<<10<<endl;
```

} 10 lần lệnh cout



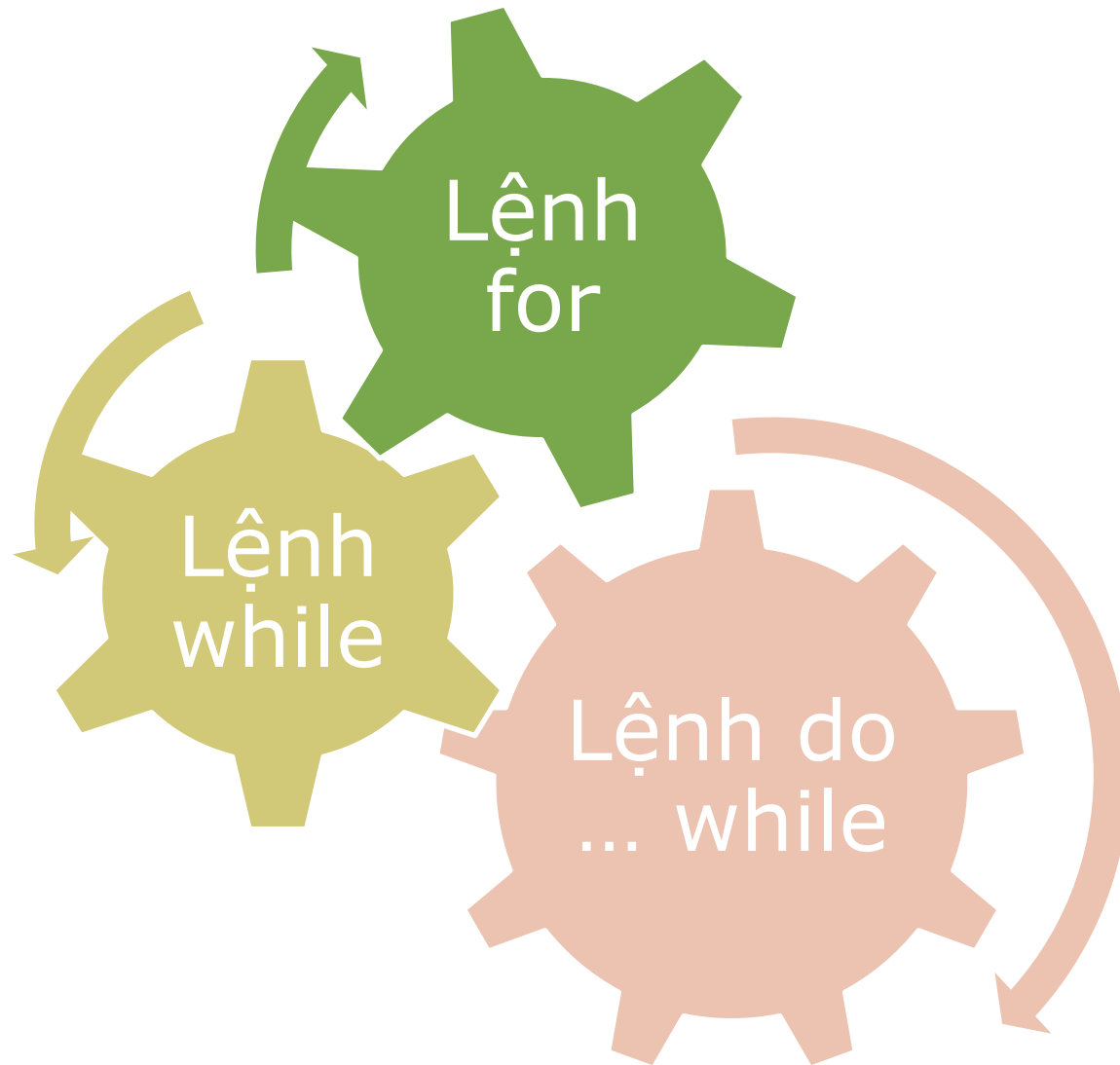


# Chương 5 Câu lệnh lặp

	Lưu đồ	Thứ tự thực hiện
<b>Dạng 1</b>	<pre>graph TD; Entry(( )) --&gt; DK{Điều kiện ?}; DK -- đúng --&gt; V1[Việc]; V1 --&gt; DK; DK -- sai --&gt; V2[Việc]; V2 --&gt; Exit(( ));</pre>	Kiểm tra điều kiện, nếu đúng thì thực hiện <việc> trong vòng lặp sau đó quay lên kiểm tra điều kiện nếu còn đúng thì thực hiện tiếp, nếu điều kiện sai thì thực hiện <việc> sau lặp.
<b>Dạng 2</b>	<pre>graph TD; Entry(( )) --&gt; V1[Việc]; V1 --&gt; DK{điều kiện?}; DK -- đúng --&gt; Entry; DK -- sai --&gt; V2[Việc]; V2 --&gt; Exit(( ));</pre>	Thực hiện <việc> trong vòng lặp rồi kiểm tra điều kiện nếu đúng thì thực hiện tiếp <việc> cho đến khi điều kiện sai thực hiện <việc> sau lặp.



# Chương 5 Câu lệnh lặp



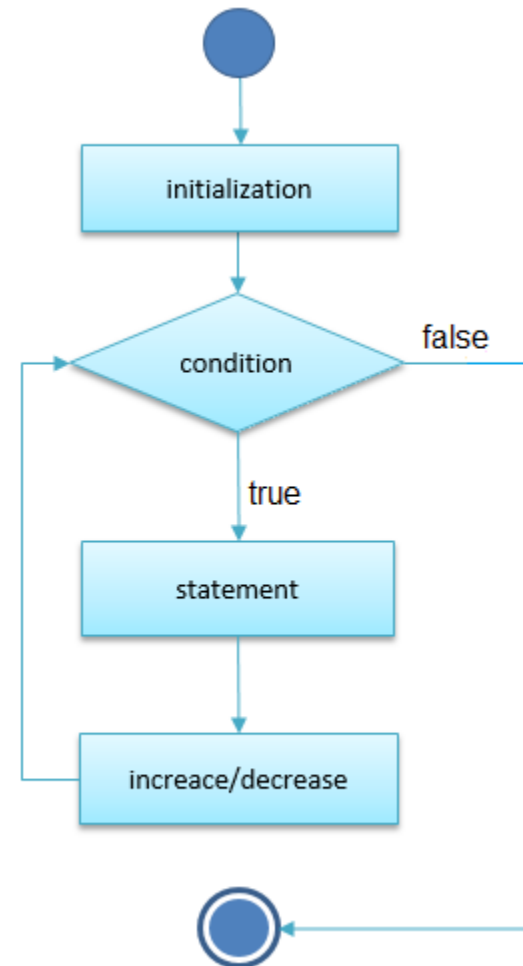


# 5.1 Lệnh for

## ❖ Cú pháp

**for** (**initialization**; **condition**; **increase/decrease**)  
**statement**;

- **initialization** (lệnh khởi tạo): khởi tạo giá trị cho biến điều khiển vòng lặp
- **condition** (điều kiện): để dừng vòng lặp
- **increase/decrease**: tăng/giảm biến điều khiển
- **statement**: lệnh trong vòng lặp





## 5.1 Lệnh for

- ❖ **Ví dụ:** In danh sách các số từ 1 đến 10 ra màn hình. Mỗi số trên một dòng.

```
for(int i=1; i <= 10; i++)  
    cout<<i<<endl;
```

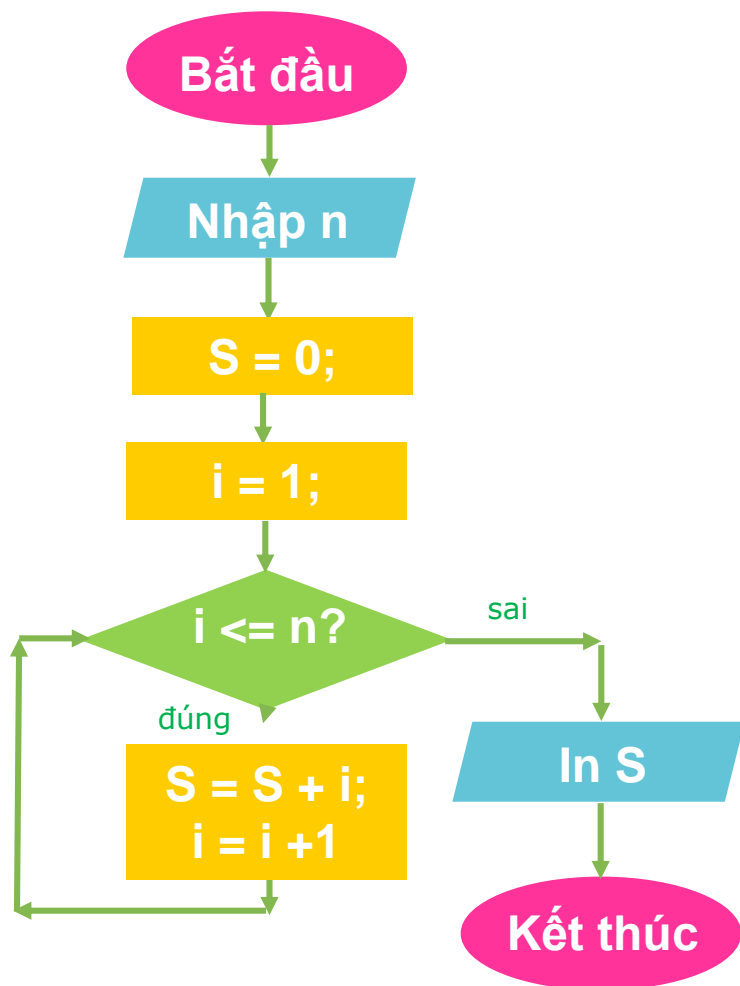
- ❖ **Ví dụ:** In danh sách các số từ 10 đến 1 ra màn hình. Mỗi số trên một dòng.

```
for(int i=10; i >= 1; i--)  
    cout<<i<<endl;
```



# 5.1 Lệnh for

**Ví dụ:** Tính tổng  $S = 1 + 2 + 3 + \dots + n$



```
1. #include <iostream.h>
2. void main()
3. {
4.     int n;
5.     long S;
6.     cout<<"Nhap n= "; cin>>n;
7.     S = 0;
8.     for (int i = 1; i <= n; i++)
9.         S = S + i;
10.    cout<<"Tong la "<<S<<endl;
11. }
```





## 5.1 Lệnh for

### ❖ Cú pháp

```
for(initialization; condition; increase/decrease)  
    statement;
```

### ❖ Lưu ý

- initialization và increase/decrease nếu có nhiều lệnh thì ngăn cách nhau bởi dấu ,
- Các đối số của lệnh for có thể thiếu nhưng dấu ; cần được giữ nguyên
- statement có từ 2 lệnh trở lên thì đặt trong cặp dấu { ... }



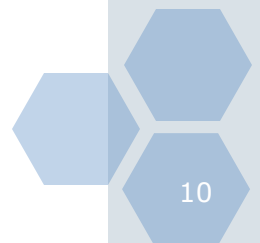


## 5.1 Lệnh for

**Ví dụ:** Tính tổng  $S = 1 + 2 + 3 + \dots + n$

```
1. #include <iostream.h>
2. void main()
3. {
4.     int i, n;
5.     long S;
6.     cout<<"Nhap n= "; cin>>n;
7.     for (i = 1, S = 0; i <= n; i++)
8.         S = S + i;
9.     cout<<"Tong la "<<S<<endl;
10. }
```

Lệnh khởi tạo có 2 lệnh ngăn cách nhau bởi dấu ,





## 5.1 Lệnh for

**Ví dụ:** Tính tổng  $S = 1 + 2 + 3 + \dots + n$

```
1. #include <iostream.h>
2. void main()
3. {
4.     int i = 1, n;
5.     long S = 0;
6.     cout<<"Nhap n= ";
7.     for (; i <= n;)
8.     {
9.         S = S + i;
10.        i = i + 1;
11.    }
12.    cout<<"Tong la "<<S<<endl;
```

Vòng for khuyết lệnh khởi tạo, lệnh tăng/giảm nhưng dấu ; vẫn được giữ nguyên

Lệnh trong vòng lặp có từ 2 lệnh trở lên đặt trong cặp dấu { ... }



# 5.1 Lệnh for

## ❖ Các lỗi thường gặp khi viết lệnh for

```
for (count=0; count<totalNumbers;)
{
    cin>>aNum;
    sum += aNum;
}
```

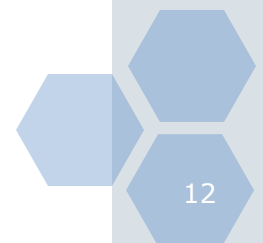
Biến điều khiển không được cập nhật sau mỗi lần lặp

```
for (count=0; count<totalNumbers;count++);
{
    cin>>aNum;
    sum += aNum;
}
```

Không có dấu ; ở đây

```
for (count=0, count<totalNumbers,count++)
{
    cin>>aNum,
    sum += aNum;
}
```

Không phải dấu , mà là dấu ;

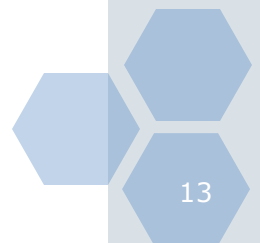




## 5.1 Lệnh for

**Ví dụ:** Viết chương trình tính tổng  $S = 2 + 4 + 6 + \dots + 2n$

```
1. #include <iostream.h>
2. int main()
3. {
4.     int i,n;
5.     long S = 0;
6.     cout<<"Nhap vao so n: ";cin>>n;
7.     for(i=1; i<=n; i++)
8.         S = S + 2*i;
9.     cout<<"Tong S = "<<S<<endl;
10.
11.     return 0;
12. }
```



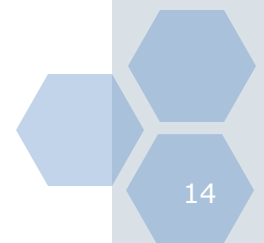
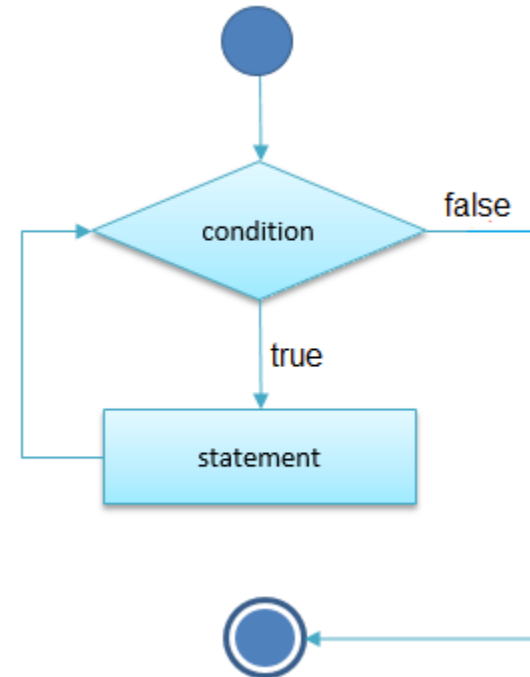


## 5.2 Lệnh while

### ❖ Cú pháp

```
while (condition)  
    statement;
```

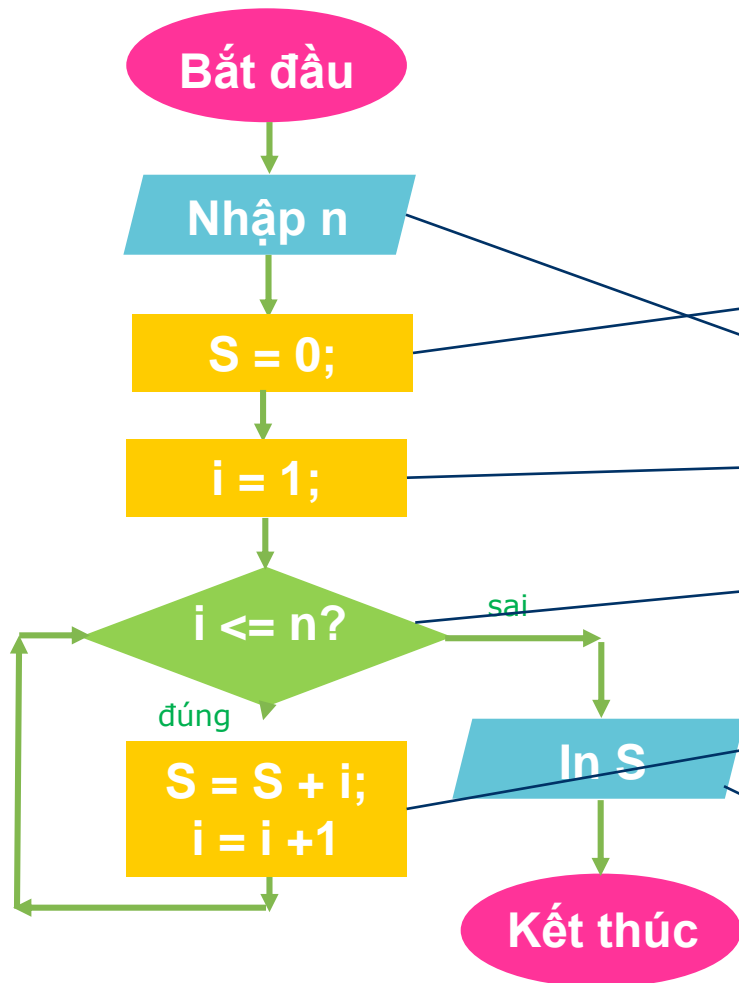
- Vòng lặp while không tự động khởi tạo giá trị biến điều khiển
- Và không tự động tăng giá trị của biến điều khiển
- Dùng cho vòng lặp không biết trước số lần lặp





## 5.2 Lệnh while

**Ví dụ:** Tính tổng  $S = 1 + 2 + 3 + \dots + n$



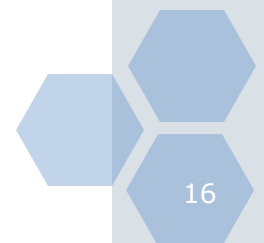
```
1. #include <iostream.h>
2. void main()
3. {   int   i, n;
4.     long S = 0;
5.     cout<<"Nhap n= "; cin>>n;
6.     i = 1;
7.     while (i <= n)
8.     {       S = S + i;
9.           i = i + 1;
10.    }
11.    cout<<"Tong la "<<S<<endl;
12. }
```



## 5.2 Lệnh while

**Ví dụ:** Viết chương trình tính tổng  $S = 2 + 4 + 6 + \dots + 2n$

```
1. #include <iostream.h>
2. int main()
3. {
4.     int i = 2, n;
5.     long S = 0;
6.     cout<<"Nhap vao so n: "; cin>>n;
7.     while (i<=2*n)
8.     {
9.         S = S + i;
10.        i = i + 2;
11.    }
12.    cout<<"Tong S = "<<S<<endl;
13.    return 0;
14. }
```





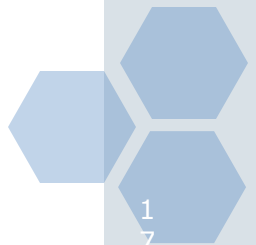


## Câu hỏi củng cố bài

1. Lệnh nào sau đây không phải là cấu trúc lặp trong C++?
  - A. for
  - B. while
  - C. do ... while
  - D. repeat ...until



Multiple Choice



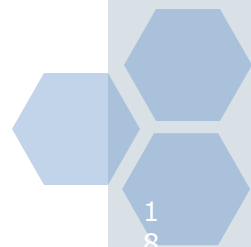


## Câu hỏi củng cố bài

2. Khi nào lệnh **while** ( $x < 100$ ) thực thi?
- A. Khi  $x$  nhỏ hơn 100
  - B. Khi  $x$  lớn hơn 100
  - C. Khi  $x$  bằng 100
  - D. Khi  $x$  lớn hơn bằng 100



Multiple Choice





## Câu hỏi củng cố bài

3. Giá trị cuối cùng của x khi đoạn chương trình được chạy là bao nhiêu?

A. 9

```
int x;
```

B. 10

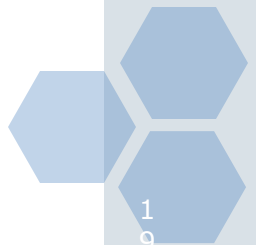
```
for (x=0; x<10; x++) { }
```

C. 0

D. 1



Multiple Choice





## Câu hỏi củng cố bài

4. Đoạn lệnh sau cho kết quả thể nào?

A. 10    11    12    13    14

B. 11    12    13    14    15

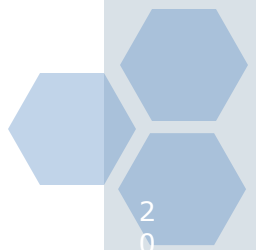
C. 10    11    12    13    14    15

D. 11    12    13    14

```
1 #include <iostream.h>
2 int main () {
3     int a;
4
5     a = 10;
6     for (; a < 15; ) {
7         cout<<"\t"<<a;
8         a++;
9     }
10    return 0;
11 }
```



Multiple Choice





## Câu hỏi củng cố bài

5. Đoạn lệnh sau cho kết quả thể nào?

A. 1 2 3 4 5

B. 0 2 4 6 8 10

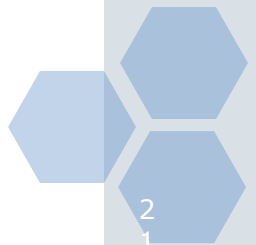
C. 2 4 6 8 10

D. 0 2 4 6 8

```
1 #include <iostream.h>
2 int main()
3 {
4     int i;
5     for(i = 1; i <= 10; i++)
6         if(i%2 == 0)
7             cout<<i<<"\t";
8
9     return 0;
10 }
```



Multiple Choice





## Câu hỏi củng cố bài

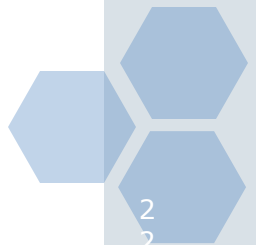
6. Đoạn lệnh sau cho kết quả thể nào?

- A. ABCDEF
- B. AAAAAABCDEF
- C. AAAAAABCDE
- D. AAAAAABCDEF

```
1 #include <iostream.h>
2 int main()
3 {
4     int i = 0;
5     char ch = 'A';
6     while(i++ < 5 || ++ch <= 'F')
7     {
8         putchar(ch);
9     }
10
11     return 0;
12 }
```



Multiple Choice





## Câu hỏi củng cố bài

7. Đoạn lệnh sau cho kết quả thể nào?

A. Hello

B. Không in gì

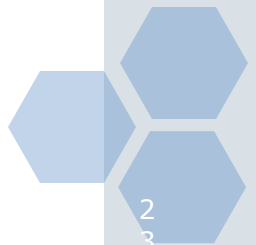
C. HelloHello

D. Báo lỗi cú pháp

```
1 #include <iostream.h>
2 int main()
3 {
4     int k;
5     for (k = -3; k < -5; k++)
6         cout<<"Hello";
7     return 0;
8 }
```



Multiple Choice





## Câu hỏi củng cố bài

8. Đoạn lệnh sau cho kết quả thể nào?

A. Hello

B. Báo lỗi cú pháp

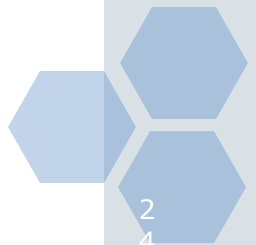
C. HelloHello

D. Không in gì

```
1 #include <iostream.h>
2 int main()
3 {
4     int k;
5     for (k = -5; k < -3; k++)
6         cout<<"Hello";
7     return 0;
8 }
```



Multiple Choice







## Câu hỏi củng cố bài

9. Đoạn lệnh sau cho kết quả thể nào?

A. True

B. False

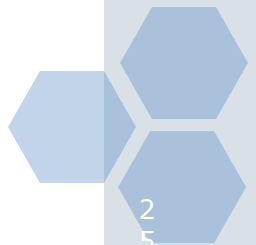
C. Báo lỗi cú pháp

D. TrueFalse

```
1 #include <iostream.h>
2 int main()
3 {
4     int i = 0;
5     while (i = 0)
6         cout<<"True\n";
7     cout<<"False\n";
8     return 0;
9 }
```

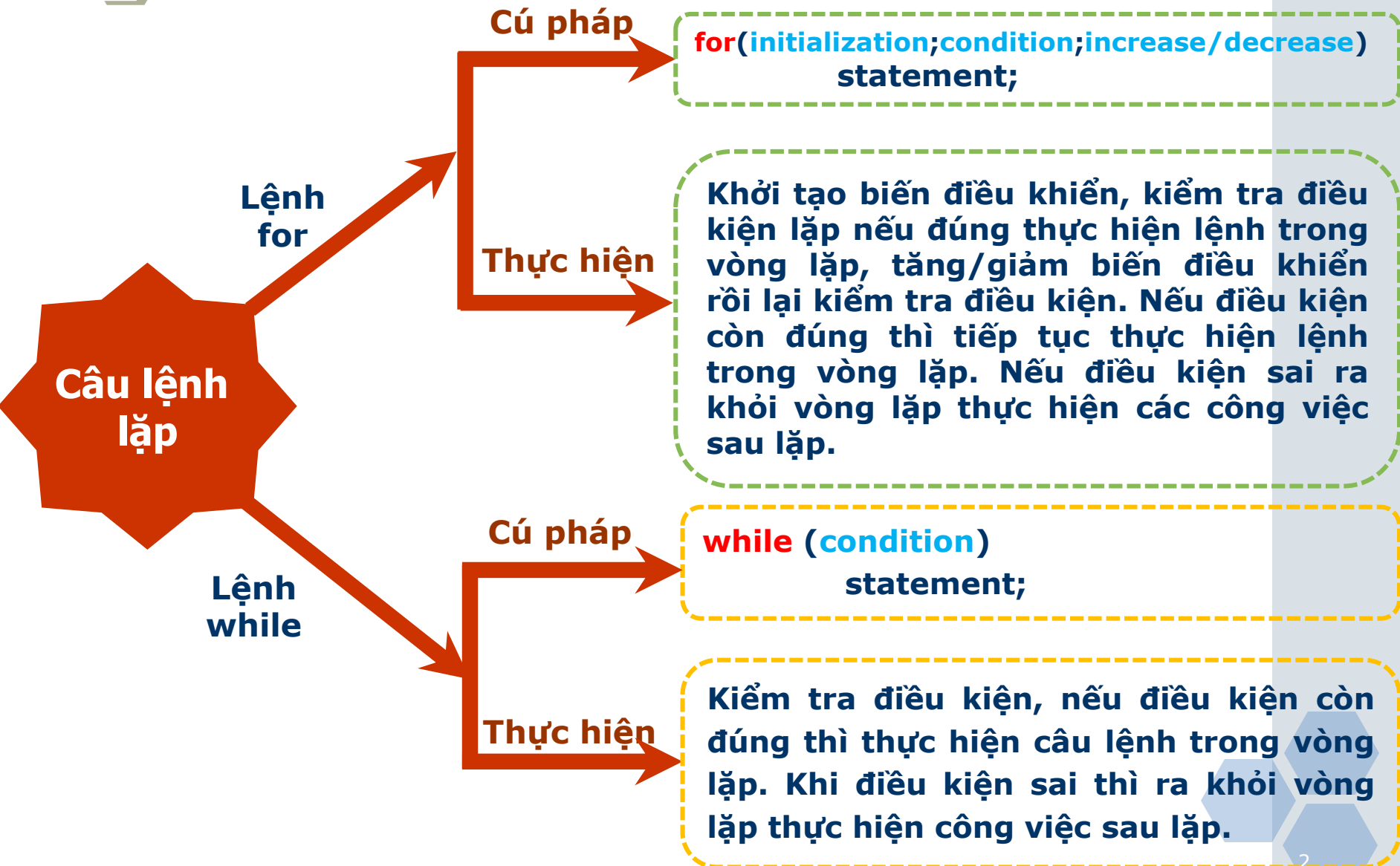


Multiple Choice





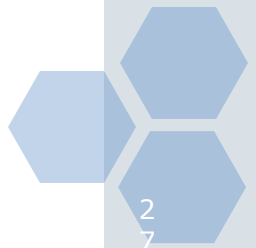
# Tổng kết





# Câu hỏi lý thuyết

1. Nêu cú pháp câu lệnh for? Cho ví dụ.
2. Nêu cú pháp câu lệnh while? Cho ví dụ.
3. Phân biệt lệnh for và lệnh while.



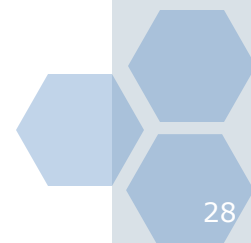


## Bài tập

Viết chương trình C++ thực hiện các công việc sau:

1. Nhập vào số nguyên dương  $n$ . In ra tất cả các ước số của  $n$ .
2. Nhập vào số nguyên dương  $n$ . In ra các số nguyên tố trong phạm vi  $n$ .
3. Nhập vào số nguyên dương  $n$ . In ra các số hoàn hảo trong phạm vi  $n$ . Biết số hoàn hảo là số có tổng các ước số không kể chính nó bằng chính số đó.

Ví dụ:  $1 + 2 + 3 = 6 \Rightarrow 6$  là số hoàn hảo

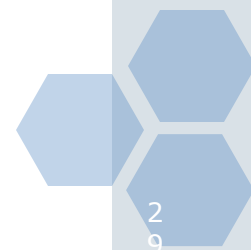




## Bài tập

Viết chương trình C++ thực hiện các công việc sau:

4. Tính tổng  $S = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$
5. Tính tích  $P = 1 * 3 * 5 * \dots * (2n-1)$
6. Nhập vào một số thực cho đến khi số 0 được nhập vào:
  - a) Đếm số phần tử đã nhập, không kể số 0
  - b) Tính trung bình cộng các phần tử đã nhập
  - c) Trong đó có bao nhiêu phần tử âm





## Bài tập

Viết chương trình C++ thực hiện các công việc sau:

7. Nhập vào một ký tự cho đến khi dấu \* được nhập vào:
  - a) Đếm số ký tự đã nhập, không kể dấu \*
  - b) Trong đó có bao nhiêu ký tự 'k'
8. Nhập vào 2 số nguyên dương  $p$ ,  $q$ . Tìm và in ra màn hình UCLN và BCNN của  $p$  và  $q$ .

