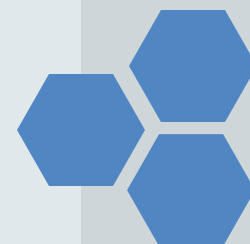


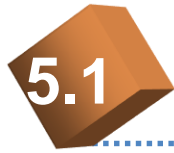
Phần 2. NN lập trình C++

CHƯƠNG 5 CÂU LỆNH LẶP (tt)





Chương 5 – Câu lệnh lặp



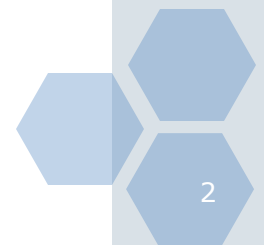
Lệnh for



Lệnh while



Lệnh do ... while



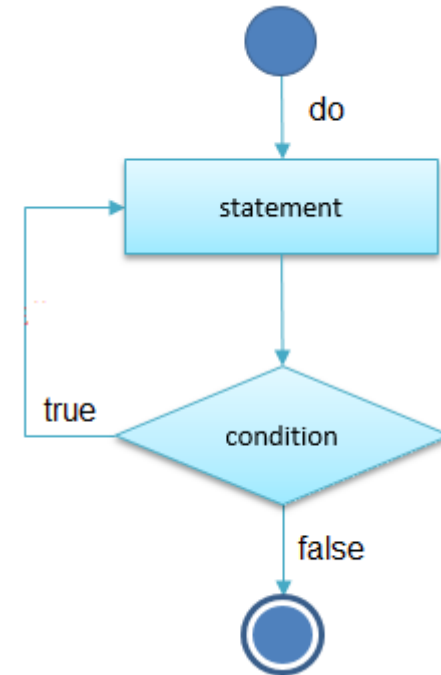


5.3 Lệnh do ... while

❖ Cú pháp

```
do {  
    statement;  
} while (condition)
```

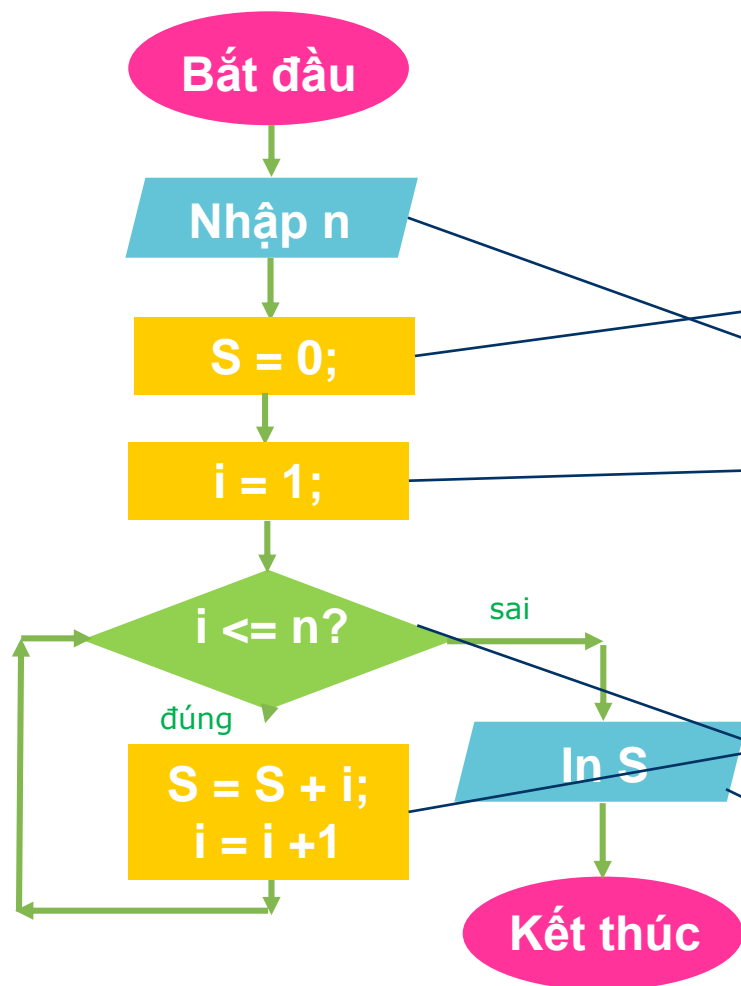
- Vòng lặp do ... while không tự động khởi tạo giá trị biến điều khiển
- Và không tự động tăng giá trị của biến điều khiển
- Dùng cho vòng lặp không biết trước số lần lặp





5.3 Lệnh do ... while

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



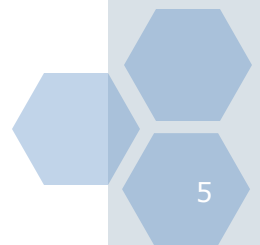
```
1. #include <iostream.h>
2. void main()
3. {   int   i, n;
4.     long S = 0;
5.     cout<<"Nhap n= "; cin>>n;
6.     i = 1;
7.     do
8.     {       S = S + i;
9.           i++;
10.    } while (i <= n);
11.     cout<<"Tong la "<<S<<endl;
12. }
```

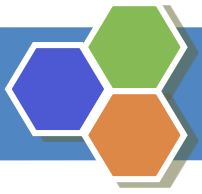


5.3 Lệnh do ... while

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

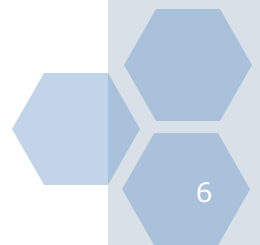
```
1. #include <iostream.h>
2. int main()
3. {
4.     int i = 2, n;
5.     long S = 0;
6.     cout<<"Nhap vao so n: "; cin>>n;
7.     do
8.     {
9.         S = S + i;
10.        i = i + 2;
11.    } while (i<=2*n);
12.    cout<<"Tong S = "<<S<<endl;
13.    return 0;
14. }
```





Lệnh break và continue

- **break**
 - Thoát ngay ra khỏi các cấu trúc **while**, **for**, **do/while**, **switch**
 - Chương trình tiếp tục chạy tại lệnh đầu tiên ngay sau cấu trúc
- thường được sử dụng để
 - Thoát khỏi vòng lặp sớm hơn bình thường
 - bỏ qua phần còn lại của **switch**



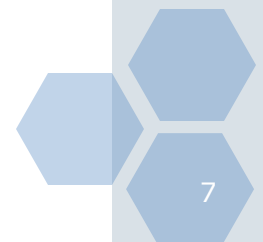


Lệnh break và continue

```
1. #include <iostream.h>
2
3 // ham than chuong trinh
4 int main()
5 {
6
7     int x;
8
9     // lap 10 lan
10    for ( x = 1; x <= 10; x++ ) {
11
12        if ( x == 5 )
13            break;
14
15        cout << x << " ";
16
17    } // ket thuc vong for
18
19    cout << "\nBroke out of loop when x became " << x << endl;
20
21    return 0;
22
23 } // ket thuc ham main
```

Thoát khỏi vòng **for** khi **break** được thực thi.

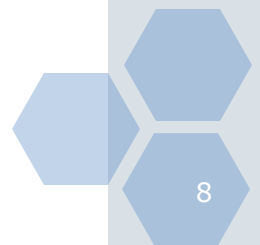
```
1 2 3 4
Broke out of loop when x became 5
```





Lệnh **break** và **continue**

- **continue**
 - được dùng trong **while**, **for**, **do/while**
 - bỏ qua phần còn lại của thân vòng lặp
 - chạy tiếp lần lặp tiếp theo
- với các vòng **while** và **do/while**
 - thực hiện kiểm tra điều kiện lặp ngay sau lệnh **continue**
- với vòng **for**
 - biểu thức tăng/giảm biến đếm được thực hiện
 - sau đó, điều kiện lặp được kiểm tra





Lệnh break và continue

```
1  #include <iostream.h>
2
3  // function main begins program
4  int main()
5  {
6  // loop 10 times
7  for ( int x = 1; x <= 10; x++ ) {
8
9  // if x is 5, continue with next iteration of loop
10     if ( x == 5 )
11         continue;    // skip remaining code in loop body
12
13     cout << x << " "; // display value of x
14
15 } // end for structure
16
17 cout << "\nUsed continue to skip printing the value 5"
18     << endl;
19
20 return 0;    // indicate successful termination
21
22 } // end function main
```

Bỏ qua phần còn lại của thân vòng **for**, nhảy đến lần lặp tiếp theo.

```
1 2 3 4 6 7 8 9 10
Used continue to skip printing the value 5
```



So sánh while và do ... while

```
#include <iostream.h>
int main()
{
    int num, sum = 0;
    int i = 0, n;
    cout<<"Nhap n: ";cin>>n;
    while (i < n)
    {
        cin>>num;
        sum += num;
        i++;
    }
    cout<<"Sum is "<<sum<<endl;
    return 0;
}
```

```
#include <iostream.h>
int main()
{
    int num, sum = 0;
    int i = 0, n;
    cout<<"Nhap n: ";cin>>n;
    do {
        cin>>num;
        sum += num;
        i++;
    } while (i < n);
    cout<<"Sum is "<<sum<<endl;
    return 0;
}
```



Câu hỏi củng cố bài

1. Đoạn lệnh sau cho kết quả thể nào?

A. 1,2,3,4,5,

B. 1,2,4,5,

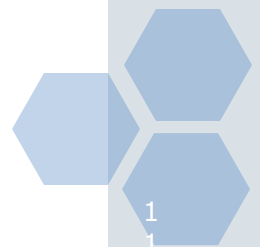
C. Báo lỗi cú pháp

D. 1,2,

```
for(int i = 1; i < 5; i++) {  
    if(i == 3) break;  
    cout << i << ",";  
}
```



Multiple Choice





Câu hỏi củng cố bài

2. Đoạn lệnh sau cho kết quả thể nào?

A. 1,2,3,4,

B. 1,2,4,

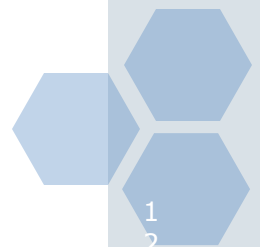
C. Báo lỗi cú pháp

D. 1,2,

```
for(int i = 1; i < 5; i++) {  
    if(i == 3) continue;  
    cout << i << “,”;  
}
```



Multiple Choice





Câu hỏi củng cố bài

3. Đoạn lệnh sau cho kết quả thể nào?

A. 0,

B. 1,2,3

C. Lặp vĩnh viễn

D. Báo lỗi cú pháp

```
int x = 0;
```

```
do {
```

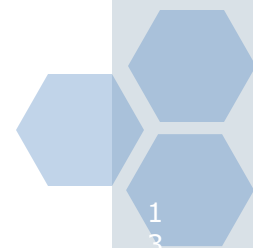
```
    cout << x << " ";
```

```
    --x;
```

```
} while (x > 0);
```



Multiple Choice





Câu hỏi củng cố bài

4. Đoạn lệnh sau cho kết quả thể nào?

A. 3,2,1,

B. Lặp vĩnh viễn

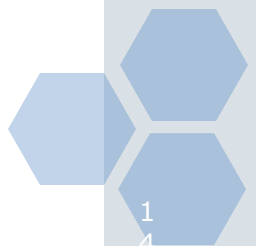
C. Lỗi cú pháp

D. 1,2,3,

```
int x = 3;  
do {  
    cout << x << " ";  
    --x;  
} while (x > 0)
```



Multiple Choice





Câu hỏi củng cố bài

5. Đoạn lệnh sau cho kết quả thể nào?

A. 1,2,3,

B. 3,2,1,

C. Báo lỗi cú pháp

D. Lặp vĩnh viễn

```
int x = 3;
```

```
do {
```

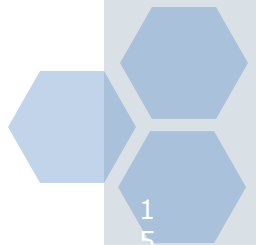
```
    cout << x << " ";
```

```
    --x;
```

```
} while (x > 0);
```



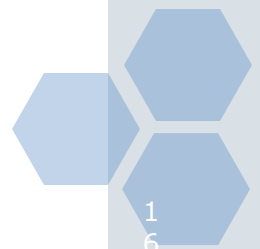
Multiple Choice





Câu hỏi lý thuyết

1. Nêu cú pháp câu lệnh do ... while? Cho ví dụ.
2. Phân biệt lệnh break và lệnh continue. Cho ví dụ minh họa.

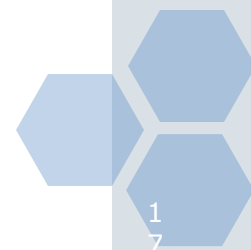




Bài tập

Viết chương trình C++ thực hiện các công việc sau dùng lệnh do ... while:

1. Tính tổng $S = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$
2. Tính tích $P = 1 * 3 * 5 * \dots * (2n-1)$
3. 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

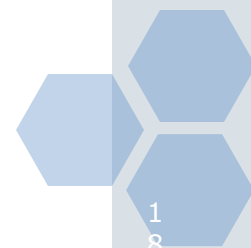




Bài tập

Viết chương trình C++ thực hiện các công việc sau dùng lệnh do ... while:

4. 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'
5. Nhập vào 2 số nguyên dương p, q. Tìm và in ra màn hình UCLN và BCNN của p và q.





Bài tập

Viết chương trình C++ thực hiện các công việc sau dùng lệnh do ... while:

6. Viết chương trình nhập vào một số thực a ($a < 2$). Tìm số n nhỏ nhất sao cho

$$S = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} > a$$

Với n tìm được hãy in ra S

