

KỸ THUẬT LẬP TRÌNH C/C++

Chương 4: CẤU TRÚC ĐIỀU KHIỂN Rẽ NHÁNH

Nội dung

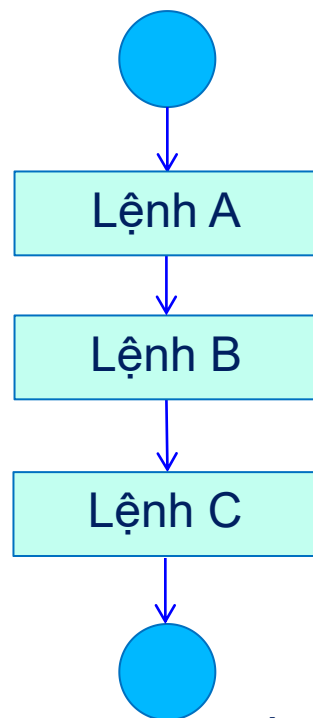
1. Tổng quan cấu trúc điều khiển
2. Mệnh đề if else
3. Mệnh đề switch case
4. Toán tử điều kiện

1. Tổng quan cấu trúc điều khiển

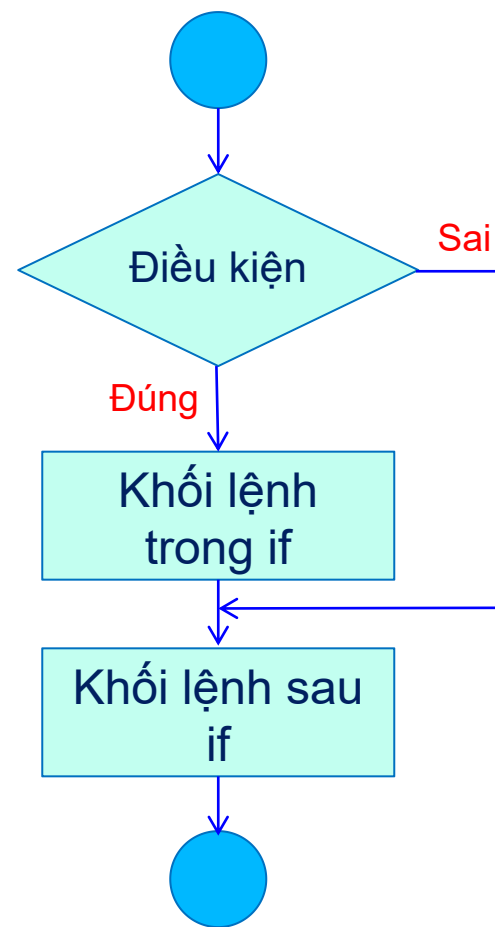
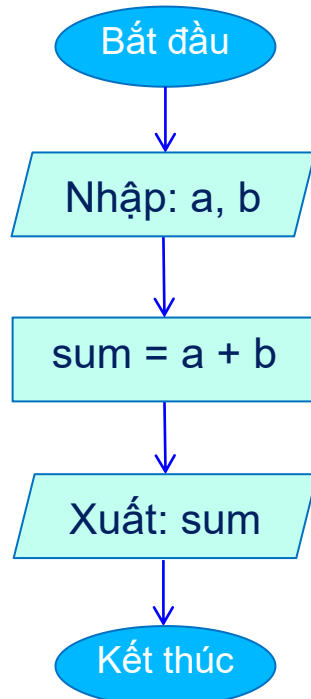
1. Tổng quan cấu trúc điều khiển

Tuỳ theo yêu cầu của bài toán mà người lập trình C++ có thể sử dụng một trong ba loại cấu trúc xử lý như sau:

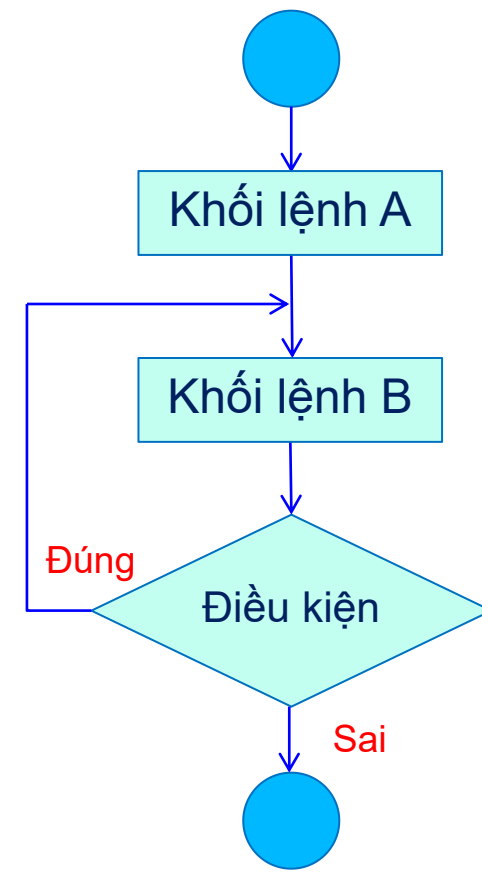
1. Cấu trúc tuần tự.
2. Cấu trúc điều khiển rẽ nhánh (hay lựa chọn).
3. Cấu trúc điều khiển lặp.



Cấu trúc tuần tự



Cấu trúc điều khiển rẽ nhánh



Cấu trúc điều khiển lặp

1. Tổng quan cấu trúc điều khiển

❖ Cấu trúc điều khiển

Cấu trúc điều khiển được sử dụng rất nhiều trong lập trình. Cấu trúc điều khiển có 2 loại:

- Cấu trúc điều khiển rẽ nhánh sử dụng các mệnh đề như:

- + if else

- + switch case

- Cấu trúc điều khiển vòng lặp sử dụng các mệnh đề như:

- + for

- + while

- + do while

Cấu trúc điều khiển cho phép thay đổi thứ tự thực hiện các câu lệnh phụ thuộc vào điều kiện kiểm tra. Điều này giúp việc thực hiện các câu lệnh trong chương trình theo ý của người lập trình chứ không cứng nhắc là từ trên xuống dưới.

1. Tổng quan cấu trúc điều khiển

❖ Khối lệnh:

Khối lệnh trong C++ là một dãy các khai báo cùng với các câu lệnh nằm trong cặp dấu ngoặc nhọn **{ }** được gọi là một khối lệnh. Một khối lệnh có thể chứa bên trong nó nhiều khối lệnh khác gọi là khối lệnh lồng nhau. Sự lồng nhau của các khối lệnh là không hạn chế.

Ví dụ 1:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() { —————> Bắt đầu khối lệnh
```

```
    string str1 = "Hello World!";
```

```
    cout << str1 << endl;
```

```
    return 0;
```

```
} —————> Kết thúc khối lệnh
```

} Khối lệnh

{

... các câu lệnh;

{

... các câu lệnh;

{

... các câu lệnh;

}

... các câu lệnh;

}

... các câu lệnh;

}

1. Tổng quan cấu trúc điều khiển

❖ Điều kiện trong cấu trúc điều khiển:

Các toán tử quan hệ và toán tử logic được sử dụng làm điều kiện trong các cấu trúc điều khiển rẽ nhánh. Một số toán tử quan hệ và toán tử logic thường được sử dụng:

- Nhỏ hơn: $(a < b)$, nếu a nhỏ hơn b thì kết quả của điều kiện này là true, ngược lại là false.
- Nhỏ hơn hoặc bằng: $(a \leq b)$, nếu a nhỏ hơn hoặc bằng b thì kết quả của điều kiện này là true, ngược lại là false.
- Lớn hơn: $(a > b)$, nếu a lớn hơn b thì kết quả của điều kiện này là true, ngược lại là false.
- Lớn hơn hoặc bằng: $(a \geq b)$, nếu a lớn hơn hoặc bằng b thì kết quả của điều kiện này là true, ngược lại là false.
- Bằng: $(a == b)$, nếu a bằng b thì kết quả của điều kiện này là true, ngược lại là false.
- Không bằng: $(a != b)$, nếu a khác b thì kết quả của điều kiện này là true, ngược lại là false.
- Các toán tử logic: $\&\&$; $\|\$, $!$

2. Mệnh đề if else

2. Mệnh đề if else

Mệnh đề if else trong C++ được sử dụng để kiểm tra một biểu thức điều kiện nào đó có đúng hay không, nếu đúng (true hay bằng 1) thì thực thi những câu lệnh bên trong khối lệnh if, ngược lại nếu sai (false hay bằng 0) thì sẽ bỏ qua những câu lệnh bên trong khối lệnh if.

Có ba dạng của câu lệnh if trong C++:

- Mệnh đề if
- Mệnh đề if - else
- Mệnh đề if - elseif - else

Ngoài ra, còn có dạng mệnh đề if lồng nhau.

Lưu ý: các mệnh đề if else, if - elseif - else đều viết bằng chữ thường.

2. Mệnh đề if else

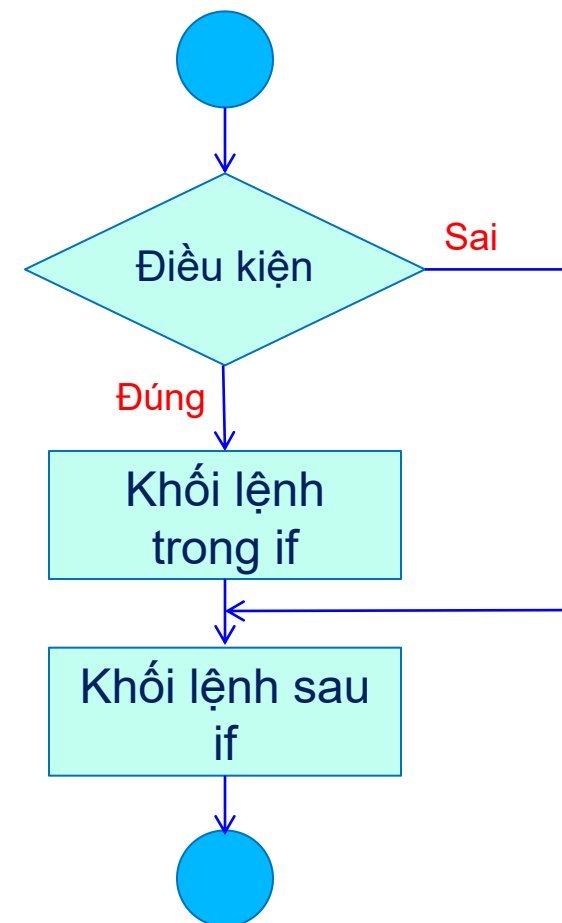
1. Mệnh đề if

Mệnh đề if được sử dụng để kiểm tra giá trị dạng boolean (true hoặc false) của điều kiện.

Cú pháp:

```
if (điều kiện) {  
    //khối lệnh trong if.  
}  
  
//khối lệnh sau if
```

- Nếu <điều kiện> đúng, khối lệnh trong if sẽ được thực thi và sau đó thực thi khối lệnh sau if.
- Nếu <điều kiện> sai, sẽ bỏ qua khối lệnh trong if, thực thi khối lệnh sau if.



2. Mệnh đề if else

1. Mệnh đề if (tt)

Ví dụ 2: Viết chương trình nhập vào số nguyên từ bàn phím trong lúc chạy chương trình. Kiểm tra số nguyên nếu số chẵn thì in ra màn hình “So chan”, nếu số lẻ thì in ra “So le”.

```
#include <iostream>
```

```
using namespace std;
```

```
int main () {
```

```
    int a;
```

```
    cout << "Nhap so nguyen: ";
```

```
    cin >> a;
```

```
    if (a%2 == 0) {
```

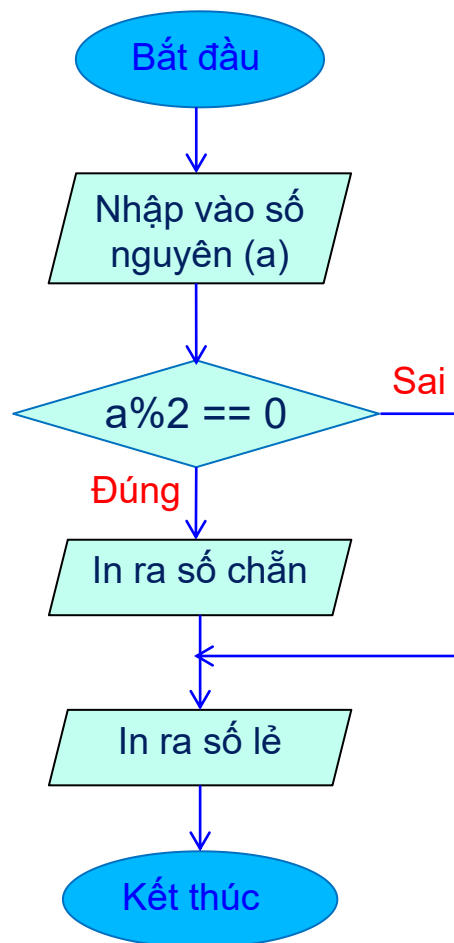
```
        cout << "So chan." << endl;
```

```
    }
```

```
    cout << "So le." << endl;
```

```
    return 0;
```

```
}
```



➤ Kết quả chạy chương trình:

- Nhập vào a = 11 ?

Số lẻ

- Nhập vào a = 12 ?

Số chẵn

Số lẻ

2. Mệnh đề if else

1. Mệnh đề if else

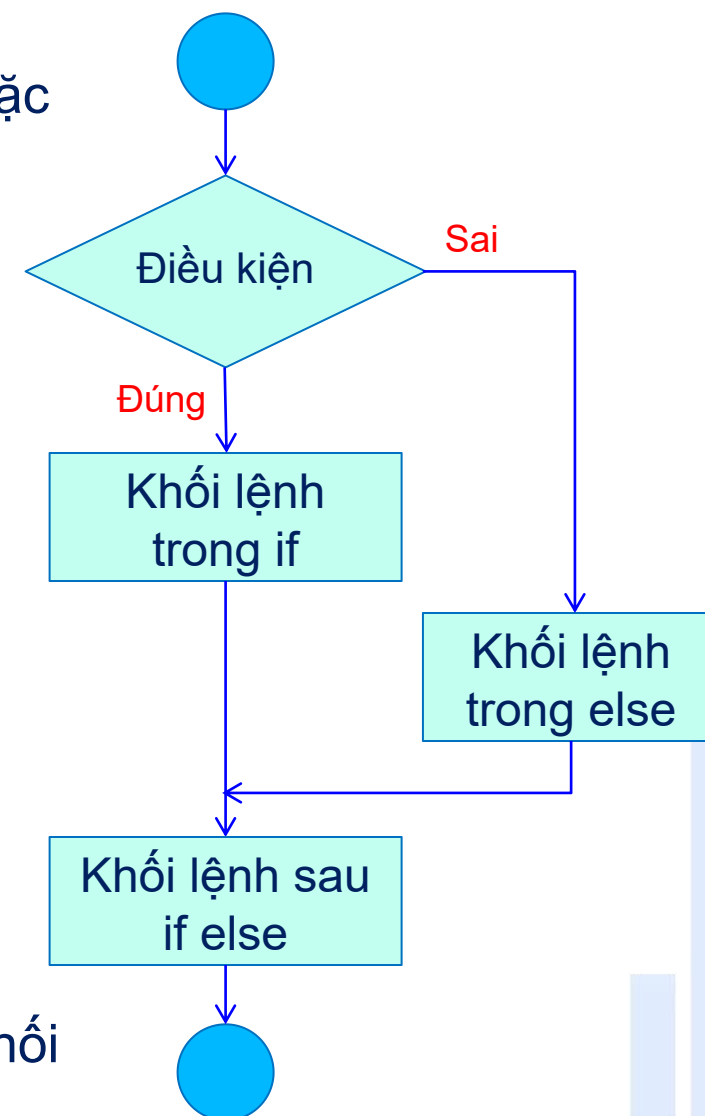
Mệnh đề if else được sử dụng để kiểm tra giá trị dạng boolean (true hoặc false) của điều kiện.

Cú pháp:

```
if (điều kiện) {  
    // khối lệnh trong if  
}  
else {  
    // khối lệnh trong else  
}
```

//khối lệnh sau if else

- Nếu <điều kiện> đúng, khối lệnh trong if sẽ được thực thi và sau đó thực thi khối lệnh sau if else.
- Nếu <điều kiện> sai, thực thi khối lệnh trong else và sau đó thực thi khối lệnh sau if else.

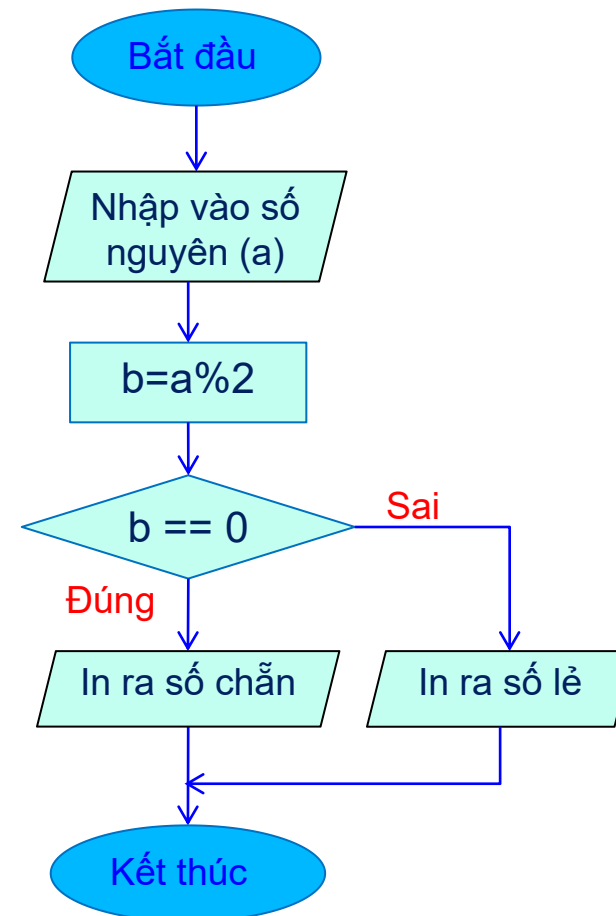
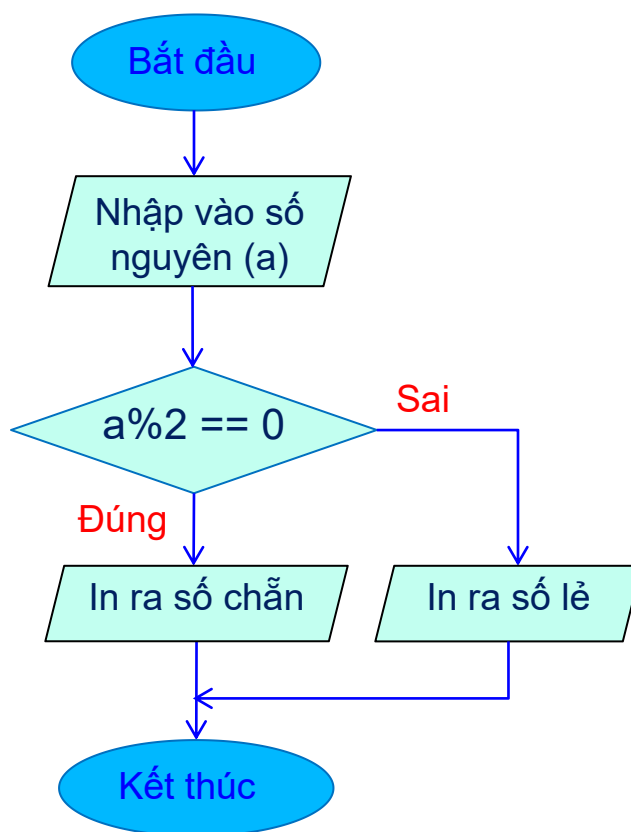


2. Mệnh đề if else

1. Mệnh đề if else (tt)

Ví dụ 3: Viết chương trình nhập vào số nguyên từ bàn phím trong lúc chạy chương trình. Kiểm tra số nguyên, nếu số chẵn thì in ra màn hình “So chan”, ngược lại thì in ra “So le”.

```
#include <iostream>
using namespace std;
int main () {
    int a;
    cout << "Nhap so nguyen: ";
    cin >> a;
    if (a % 2 == 0) {
        cout << "So chan." << endl;
    }
    else {
        cout << "So le." << endl;
    }
    return 0;
}
```



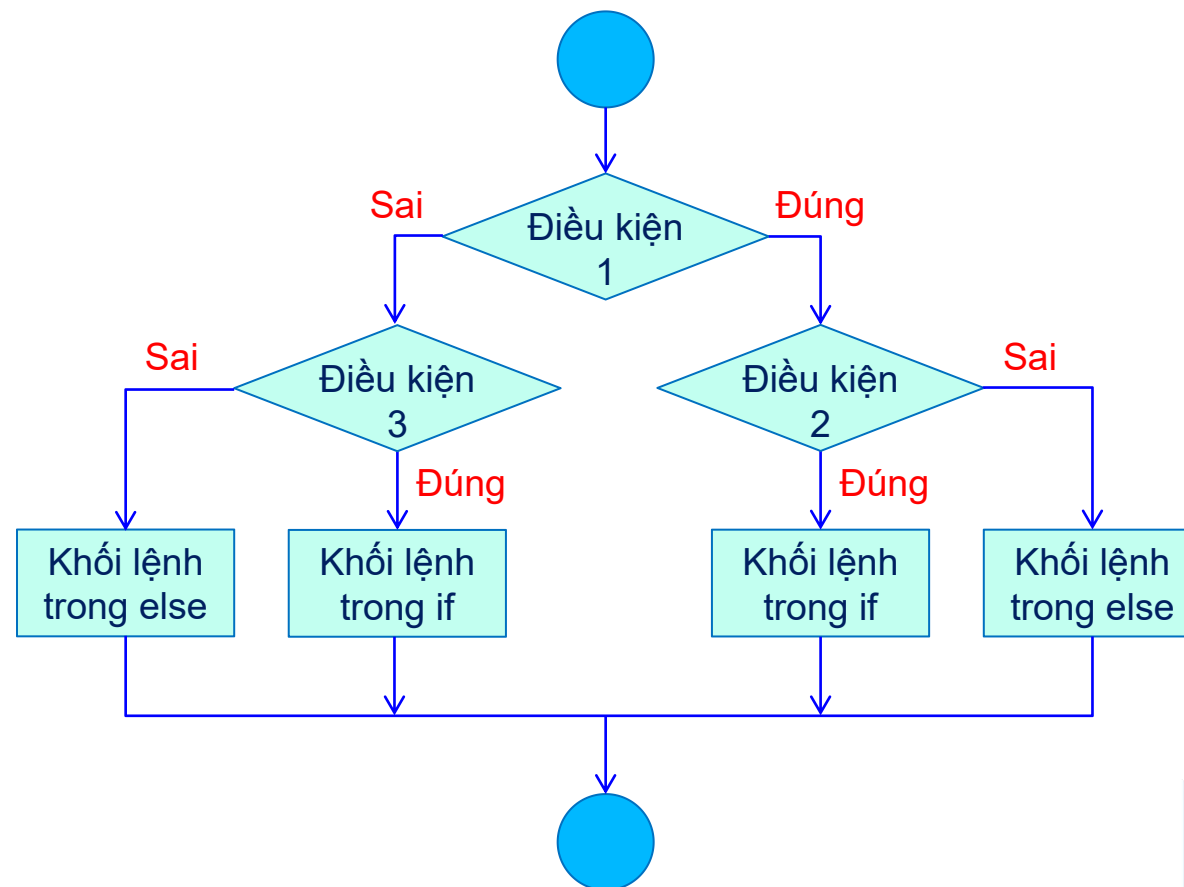
2. Mệnh đề if else

1. Mệnh đề if else (tt)

Trong khối lệnh if else có thể có các khối lệnh if else khác lồng vào.

Cú pháp:

```
if (điều kiện 1) {  
    // câu lệnh trong if với điều kiện 1  
    if (điều kiện 2) { // if trong if  
        // câu lệnh trong if với điều kiện 2  
    }  
    else {  
        // câu lệnh trong else với điều kiện 2  
    }  
}  
else { // else của điều kiện 1  
    if (điều kiện 3) { // if trong else  
        // câu lệnh trong if với điều kiện 3  
    }  
    else {  
        // câu lệnh trong else với điều kiện 3  
    }  
}
```



2. Mệnh đề if else

1. Mệnh đề if else (tt)

Ví dụ 4: Viết chương trình nhập vào số nguyên từ bàn phím trong lúc chạy chương trình. Kiểm tra số nguyên có khác không hay không?

- Nếu số nguyên khác không thì, kiểm tra:
 - + Nếu số chẵn thì in ra màn hình "So chan"
 - + Ngược lại thì in ra "So le".
- Ngược lại thì in ra màn hình là "So khong"

```
#include <iostream>
using namespace std;
int main () {
    int a;
    cout << "Nhap so nguyen: ";
    cin >> number;

    if (a != 0) {
        if (number % 2 == 0) {
            cout << "So chan." << endl;
        }
        else {
            cout << "So le." << endl;
        }
    }
    else {
        cout << "So khong." << endl;
    }
    return 0;
}
```

2. Mệnh đề if else

3. Mệnh đề if - else if - else

Mệnh đề if-elseif-else cũng kiểm tra giá trị dạng boolean của điều kiện. Nếu giá trị điều kiện if là True thì chỉ có khối lệnh trong if sẽ được thực hiện. Nếu giá trị điều kiện else if nào là True thì chỉ có khối lệnh trong if else đó sẽ được thực hiện. Nếu tất cả điều kiện của if và else if là False thì chỉ có khối lệnh sau else sẽ được thực hiện. Cú pháp:

```
if (điều kiện 1) {
```

```
    // khối lệnh này được thực thi nếu điều kiện 1 = true
```

```
} else if (điều kiện 2) {
```

```
    // khối lệnh này được thực thi nếu điều kiện 1 = false và điều kiện 2 = true
```

```
} else if (điều kiện 3) {
```

```
    // khối lệnh này được thực thi nếu điều kiện 1 và 2 = false và điều kiện 3 = true
```

```
} else {
```

```
    // khối lệnh này được thực thi nếu nếu tất cả những điều kiện trên = false
```

```
}
```


2. Mệnh đề if else

3. Mệnh đề if - else if - else (tt)

Ví dụ 5: Viết chương trình nhập vào giờ (số nguyên) từ bàn phím trong lúc chạy chương trình. Kiểm tra giờ đã nhập vào:

- Nếu giờ từ 7 đến bằng 11 thì in ra màn hình
“Chao buoi sang”
- Nếu giờ từ 12 đến bằng 13 thì in ra màn hình
“Chao buoi trưa”
- Nếu giờ từ 14 đến bằng 17 thì in ra màn hình
“Chao buoi chieu”
- Ngược lại thì in ra màn hình “Chao buoi toi”

Viết biểu thức điều kiện ?

2. Mệnh đề if else

3. Mệnh đề if - else if - else (tt)

Ví dụ 5: Viết chương trình nhập vào giờ (số nguyên) từ bàn phím trong lúc chạy chương trình. Kiểm tra giờ đã nhập vào:

- Nếu giờ từ 7 đến bằng 11 thì in ra màn hình “Chao buoi sang”
- Nếu giờ từ 12 đến bằng 13 thì in ra màn hình “Chao buoi trua”
- Nếu giờ từ 14 đến bằng 17 thì in ra màn hình “Chao buoi chieu”
- Ngược lại thì in ra màn hình “Chao buoi toi”

```
#include <iostream>
using namespace std;
int main () {
    int gio;
    cout << "Nhap vao gio: ";
    cin >> gio;
    if ((7 <= gio) && (gio <= 11)) {
        cout << "Chao buoi sang." << endl;
    }
    else if ((12 <= gio) && (gio <= 13)) {
        cout << "Chao buoi trua." << endl;
    }
    else if ((14 <= gio) && (gio <= 17)) {
        cout << "Chao buoi chieu." << endl;
    }
    else {
        cout << "Chao buoi toi." << endl;
    }
    return 0;
}
```

2. Mệnh đề if else

3. Mệnh đề if - else if - else (tt)

Ví dụ 5 (b): Viết chương trình C++ nhập vào ngày (số nguyên) từ bàn phím trong lúc chạy chương trình và in ngày trong tuần như sau (sử dụng **if - else if - else**):

- Nếu ngày bằng 1 thì in ra "Sunday".
- Nếu ngày bằng 2 thì in ra "Monday".
- Nếu ngày bằng 3 thì in ra "Tuesday".
- Nếu ngày bằng 4 thì in ra "Wednesday".
- Nếu ngày bằng 5 thì in ra "Thursday".
- Nếu ngày bằng 6 thì in ra "Friday".
- Nếu ngày bằng 7 thì in ra "Saturday".
- Ngược lại thì in ra thông báo "Khong hop le"

Viết chương trình C++ ?

2. Mệnh đề if else

3. Mệnh đề if - else if - else (tt)

Ví dụ 5 (c): Viết chương trình C++ nhập vào tháng trong năm và in ra số ngày của tháng đó.

- Nếu tháng 1 || 3 || 5 || 7 || 8 || 10 || 12 thì in ra: “ Tháng có 31 ngày”
- Nếu tháng 4 || 6 || 9 || 11 thì in ra: “ Tháng có 30 ngày”
- Nếu tháng 2 thì in ra: “ Tháng có 28 hoặc 29 ngày”
- Ngược lại in ra: “Không hợp lệ”

Viết chương trình C++ ?

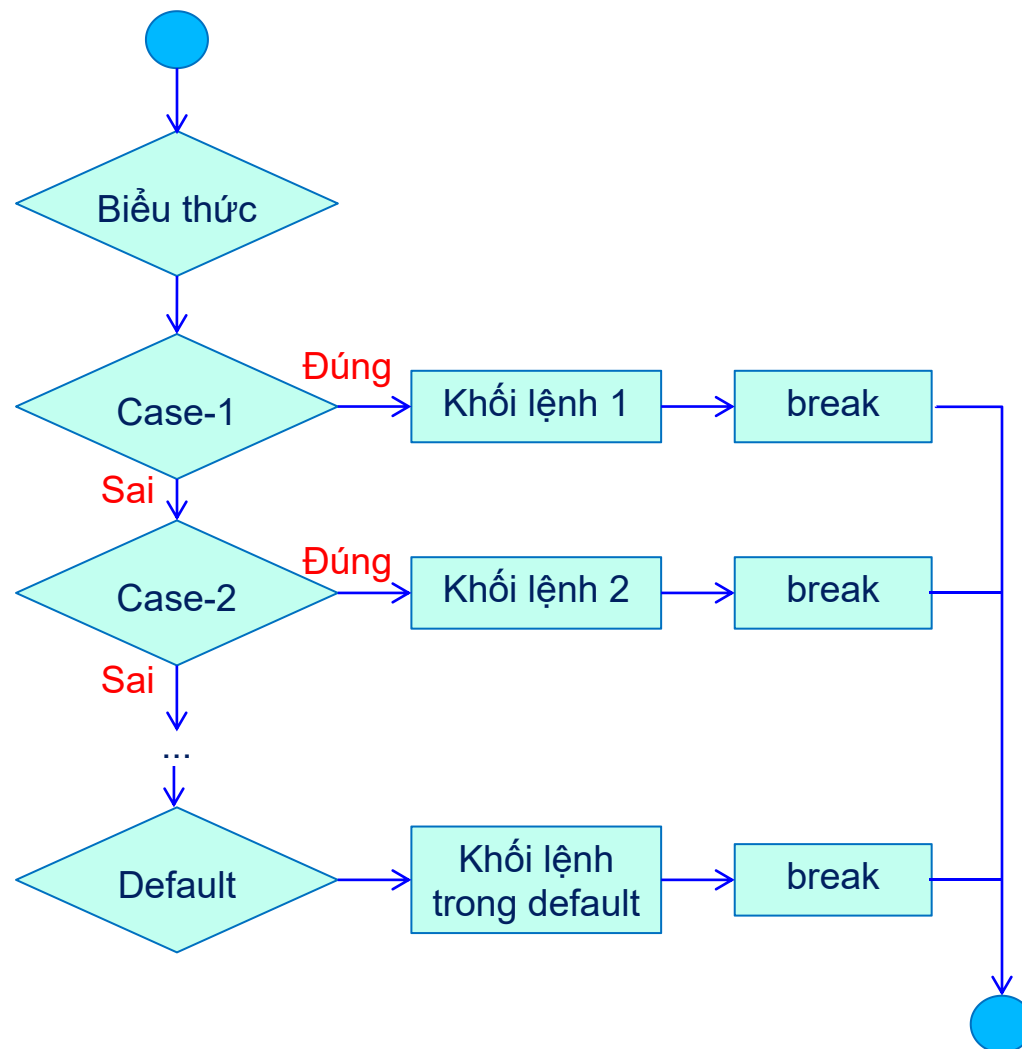
3. Mệnh đề switch case

3. Mệnh đề switch case

Mệnh đề switch case được sử dụng để thực thi một trong nhiều khối lệnh tùy vào biểu thức được lựa chọn.

Cú pháp:

```
switch (biểu thức) {  
    case giá_trị_1:  
        // khối lệnh 1  
        break;  
    case giá_trị_2:  
        // khối lệnh 2  
        break;  
    ...  
    default:  
        // khối lệnh trong default  
}
```



3. Mệnh đề switch case

Cách xử lý của mệnh đề switch case: Biểu thức trong switch được kiểm tra một lần, giá trị của biểu thức được so sánh với giá trị của mỗi trường hợp (case), nếu có sự trùng khớp, thì khối lệnh trong case đó sẽ được thực thi.

- Biểu thức trong switch có thể là một giá trị kiểu **char** hoặc **int** và phải luôn nằm trong dấu ngoặc đơn.
- **giá_trị_1, giá_trị_2,...** phải có cùng kiểu dữ liệu với giá trị của biểu thức switch.
- Từ khóa **break**; là tùy chọn, nhưng nên được sử dụng cuối mỗi trường hợp để thoát khỏi phần còn lại của lệnh switch. Nếu không có lệnh break; lệnh case tiếp theo sẽ được thực hiện.
- Trường hợp **default** là tùy chọn, được thực hiện khi không giá_trị nào đúng với biểu thức.
- Thứ tự của các giá_trị (kể cả default) là không quan trọng. Tuy nhiên, sắp xếp các giá_trị theo một trình tự logic và đặt trường hợp default cuối cùng.

3. Mệnh đề switch case

Ví dụ 6: Viết chương trình C++ nhập vào ngày (số nguyên) từ bàn phím trong lúc chạy chương trình và in ngày trong tuần như sau (sử dụng **switch case**):

```
#include <iostream>
using namespace std;
int main() {
    int ngay;
    cout << "Nhap vao ngay trong tuan: ";
    cin >> ngay;
    switch (ngay) {
        case 1:
            cout << "Sunday" << endl;
            break;
        case 2:
            cout << "Monday" << endl;
            break;
        case 3:
            cout << "Tuesday" << endl;
            break;
        case 4:
            cout << "Wednesday" << endl;
            break;
        case 5:
            cout << "Thursday" << endl;
            break;
        case 6:
            cout << "Friday" << endl;
            break;
        case 7:
            cout << "Saturday" << endl;
            break;
        default:
            cout << "Khong phai ngay trong tuan"<< endl;
    }
    return 0;
}
```


3. Mệnh đề switch case

Ví dụ 6 (b): Viết chương trình C++ nhập vào tháng trong năm và in ra số ngày của tháng đó (sử dụng **switch case**).

- Nếu tháng 1 || 3 || 5 || 7 || 8 || 10 || 12 thì in ra: “ Tháng có 31 ngày”
- Nếu tháng 4 || 6 || 9 || 11 thì in ra: “ Tháng có 30 ngày”
- Nếu tháng 2 thì in ra: “ Tháng có 28 hoặc 29 ngày”
- Ngược lại in ra: “Không hợp lệ”

Viết chương trình C++ ?

3. Mệnh đề switch case

Ví dụ 6 (c): Viết chương trình nhập vào giờ (số nguyên) từ bàn phím trong lúc chạy chương trình.

Kiểm tra giờ đã nhập vào (sử dụng **switch case**):

- Nếu giờ từ 7 đến bằng 11 thì in ra màn hình “Chao buoi sang”
- Nếu giờ từ 12 đến bằng 13 thì in ra màn hình “Chao buoi trưa”
- Nếu giờ từ 14 đến bằng 17 thì in ra màn hình “Chao buoi chieu”
- Ngược lại thì in ra màn hình “Chao buoi toi”

Viết chương trình C++ ?

4. Toán tử điều kiện

4. Toán tử điều kiện

Toán tử điều kiện được sử dụng để **thay thế cho câu lệnh if-else**.

Cú pháp:

biến_kết_quả = (điều_kiện) ? biểu_thức_1 : biểu_thức_2;

- Nếu **điều_kiện** đúng thì **biến_kết_quả = biểu_thức_1**
- Ngược lại **điều_kiện** sai thì **biến_kết_quả = biểu_thức_2**

Ví dụ 7:

```
#include <iostream>
using namespace std;
int main () {
    int number;
    cout << "Nhap so nguyen: ";
    cin >> number;
    if (number % 2 == 0) {
        cout << "So chan" << endl;
    }
    else {
        cout << "So le" << endl;
    }
    return 0;
}
```

```
#include <iostream>
using namespace std;
int main () {
    int number;
    string kq;
    cout << "Nhap so nguyen: ";
    cin >> number;
    kq = (number % 2 == 0)? "So chan" : "So le";
    cout << kq << endl;
    return 0;
}
```

4. Toán tử điều kiện

Ví dụ 8: Viết chương trình nhập vào 2 số nguyên khác nhau từ bàn phím trong lúc chạy chương trình. So sánh giữa 2 số nguyên này và in ra màn hình số nguyên nhỏ hơn.

```
#include <iostream>
using namespace std;
int main () {
    int a, b;
    cout << "Nhap so nguyen thu nhat: ";
    cin >> a;
    cout << "Nhap so nguyen thu hai: ";
    cin >> b;
    if (a < b) {
        cout << "So nho hon la: " << a << endl;
    }
    else {
        cout << "So nho hon la: " << b << endl;
    }
    return 0;
}
```

```
#include <iostream>
using namespace std;
int main () {
    int a, b, kq;
    cout << "Nhap so nguyen thu nhat: ";
    cin >> a;
    cout << "Nhap so nguyen thu hai: ";
    cin >> b;
    kq = (a < b)? a : b;
    cout << "So nho hon la: " << kq << endl;
    return 0;
}
```

4. Toán tử điều kiện

Ví dụ 9: Vẽ lưu đồ thuật toán và viết chương trình C++ tìm số nhỏ hơn giữa 2 số nguyên được nhập vào từ bàn phím trong lúc chạy chương trình và in ra số nguyên nhỏ hơn ra màn hình, trường hợp 2 số bằng nhau thì in ra thông báo “Hai số bằng nhau”.

- a) Sử dụng mệnh đề if - else
- b) Sử dụng mệnh đề if - else if - else
- c) Sử dụng toán tử điều kiện

Vẽ lưu đồ thuật toán ?

Viết chương trình C++ ?

4. Toán tử điều kiện

Ví dụ 9: Vẽ lưu đồ thuật toán và viết chương trình C++ tìm số nhỏ hơn giữa 2 số nguyên được nhập vào từ bàn phím trong lúc chạy chương trình và in ra số nguyên nhỏ hơn ra màn hình, trường hợp 2 số bằng nhau thì in ra thông báo “Hai số bằng nhau”.

a) Sử dụng mệnh đề if - else

b) Sử dụng mệnh đề if - else if - else

c) Sử dụng toán tử điều kiện

Hướng dẫn:

- Bắt đầu
- Khai báo biến a và b kiểu số nguyên
- Nhập vào 2 số nguyên, gán vào 2 biến a và b
- Kiểm tra điều kiện a bằng b hay không?
 - + Nếu đúng thì in ra thông báo “Hai số bằng nhau”
 - + Ngược lại thì kiểm tra a nhỏ hơn b hay không?
 - Nếu đúng thì in ra số nhỏ hơn là a
 - Ngược lại thì in ra số nhỏ hơn là b.
- Kết thúc