

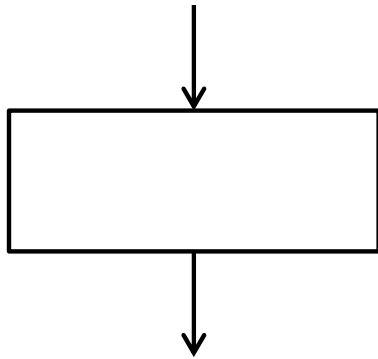
Nhập Môn Lập Trình Cấu Trúc Lựa Chọn

TS. Tô Văn Khánh
Trường Đại học Công nghệ, ĐHQGHN

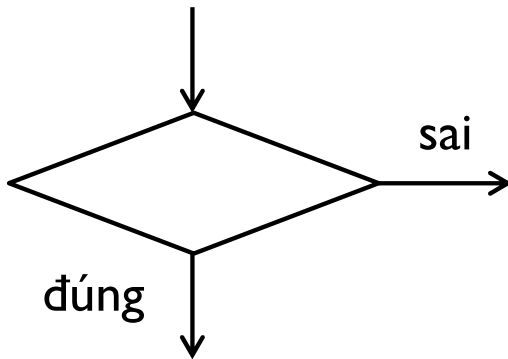
Nội Dung

- ▶ Cấu trúc điều khiển
- ▶ Cấu trúc lựa chọn:
 - ▶ **if**
 - ▶ **if ... else ...**
 - ▶ **switch**
- ▶ Biểu thức logic

Miêu Tả Chương Trình – Lưu Đồ



mệnh lệnh



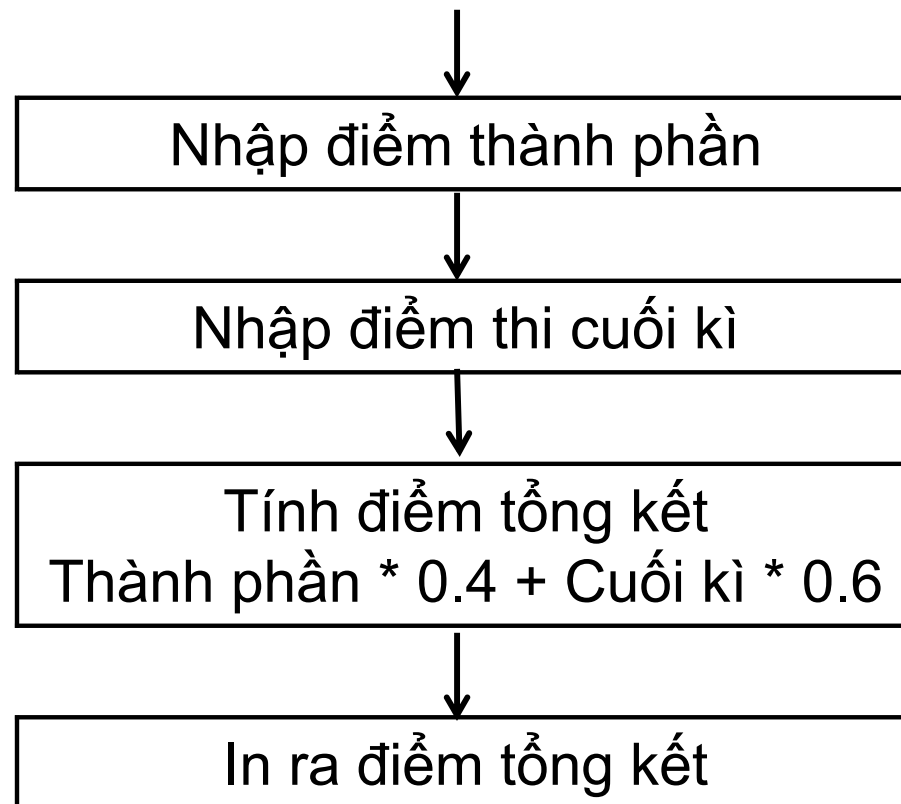
kiểm tra
điều kiện

Cấu Trúc Điều Khiển

- ▶ Thứ tự các mệnh lệnh chương trình thực hiện
- ▶ Thứ tự tuần tự:
 - ▶ thực hiện mệnh lệnh theo thứ tự trong mã nguồn
 - ▶ mệnh lệnh viết trước thực hiện trước
- ▶ Thứ tự lựa chọn:
 - ▶ mệnh lệnh thực hiện phụ thuộc điều kiện
 - ▶ cần lựa chọn mệnh lệnh nào thực hiện
 - ▶ điều khiển mã nguồn chạy như thế nào
 - ▶ ví dụ: điểm tổng kết môn dưới 4.0, in ra trượt, nếu không in ra đỗ

Cấu Trúc Điều Khiển – Tuần Tự

- ▶ Thứ tự tuần tự:
 - ▶ thực hiện mệnh lệnh theo thứ tự trong mã nguồn
 - ▶ ví dụ: tính điểm tổng kết môn học



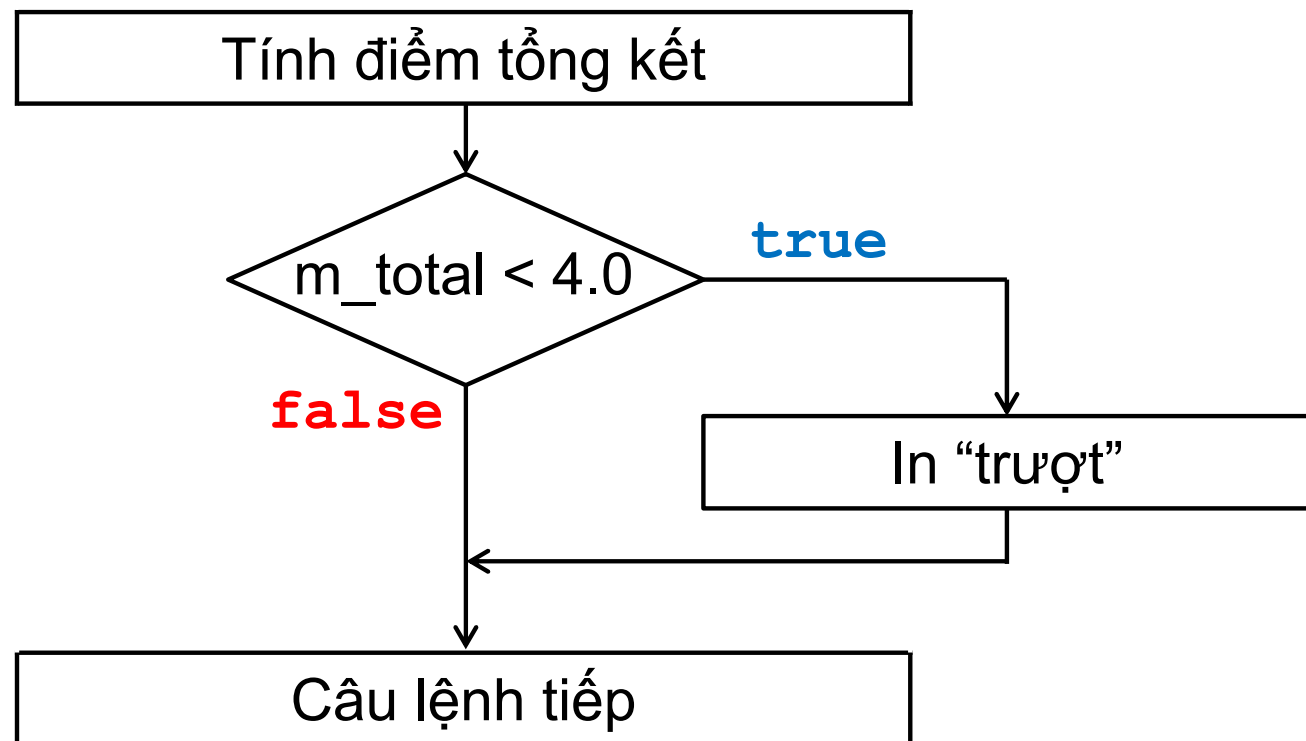
Cấu Trúc Điều Khiển – Tuần Tự

- ▶ Thứ tự tuần tự:
 - ▶ thực hiện mệnh lệnh theo thứ tự trong mã nguồn
 - ▶ ví dụ: tính điểm tổng kết môn học

```
int main()
{
    double m_mid, m_final, m_total;
    cin >> m_mid;
    cin >> m_final;
    m_total = m_mid * 0.4 + m_final * 0.6;
    cout << m_total << endl;
    return 0;
}
```

Cấu Trúc Điều Khiển – Lựa Chọn

- ▶ Thứ tự lựa chọn:
 - ▶ mệnh lệnh thực hiện phụ thuộc điều kiện
 - ▶ ví dụ: điểm tổng kết môn dưới 4.0 in ra trượt



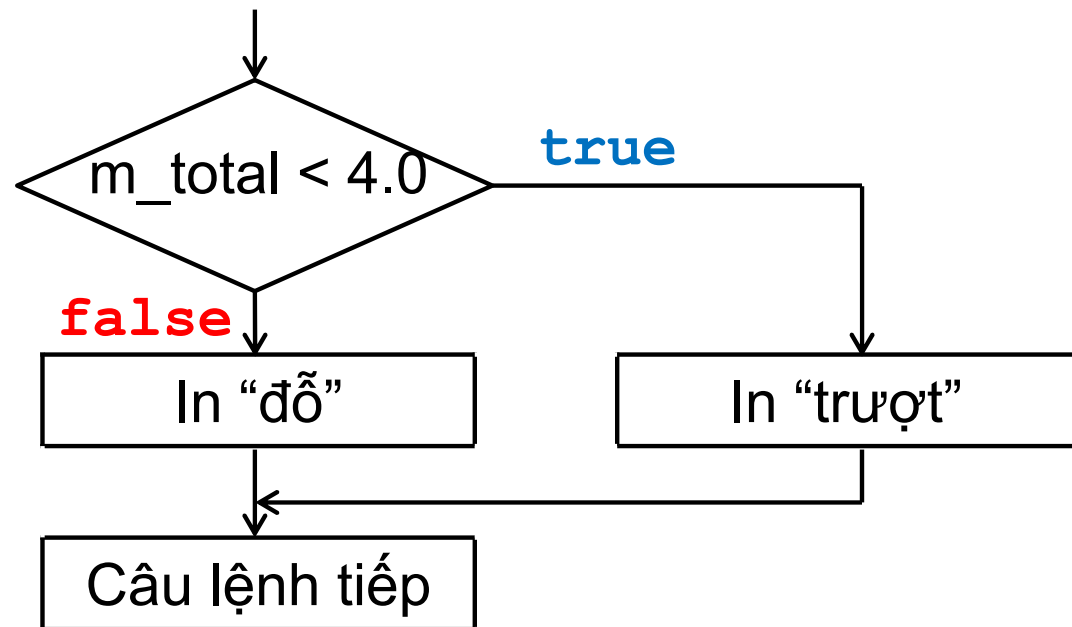
Cấu Trúc Điều Khiển – Lựa Chọn

- ▶ Thứ tự lựa chọn:
 - ▶ mệnh lệnh thực hiện phụ thuộc điều kiện
 - ▶ ví dụ: điểm tổng kết môn dưới 4.0 in ra trượt

```
int main()
{
    ... ..
    m_total = m_mid * 0.4 + m_final * 0.6;
    if (m_total < 4.0)
        cout << "truot" << endl;
    cout << m_total << endl;
    return 0;
}
```


Cấu Trúc Điều Khiển – Lựa Chọn

- ▶ Thứ tự lựa chọn:
 - ▶ mệnh lệnh thực hiện phụ thuộc điều kiện
 - ▶ ví dụ: điểm tổng kết môn dưới 4.0, in ra trượt, nếu không in ra đỗ



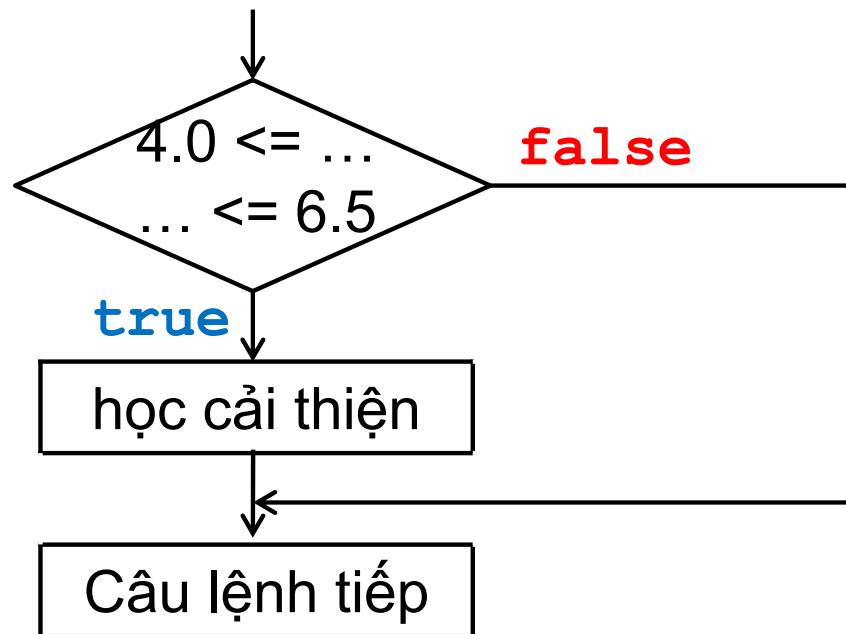
Cấu Trúc Điều Khiển – Lựa Chọn

- ▶ Thứ tự lựa chọn:
 - ▶ mệnh lệnh thực hiện phụ thuộc điều kiện
 - ▶ ví dụ: điểm tổng kết môn dưới 4.0, in ra trượt, nếu không in ra đỗ

```
int main()
{
    ... ..
    if (m_total < 4.0)
        cout << "truot" << endl;
    else
        cout << "do" << endl;
    cout << m_total << endl;
    return 0;
}
```

Cấu Trúc Điều Khiển – Lựa Chọn

- ▶ Thứ tự lựa chọn:
 - ▶ mệnh lệnh thực hiện phụ thuộc điều kiện
 - ▶ ví dụ: điểm tổng kết môn trong khoảng 4.0 đến 6.5, được phép học cải thiện ($4.0 \leq \dots \leq 6.5$)



Cấu Trúc Điều Khiển – Lựa Chọn

► Thứ tự lựa chọn:

- mệnh lệnh thực hiện phụ thuộc điều kiện
- ví dụ: điểm tổng kết môn trong khoảng 4.0 đến 6.5, được phép học cải thiện ($4.0 \leq \dots \leq 6.5$)

```
int main()
{
    ... ..
    if (m_total < 4.0) ... ..
    else ... ..
    if (4.0 <= m_total && m_total <= 6.5)
        cout << "hoc cai thien" << endl;
    cout << m_total << endl;
    return 0;
}
```

Cấu Trúc Điều Khiển – **if**

▶ Cú pháp:

```
if (biểu thức logic)  
{  
    // biểu thức logic đúng (true)  
    // thực hiện mệnh lệnh trong thân cấu trúc if  
    Mệnh Lệnh  
    Mệnh Lệnh  
}  
... ..
```

▶ Ý nghĩa:

- ▶ nếu **biểu thức logic** đúng, thân cấu trúc **if** được thực hiện
- ▶ nếu sai, điều khiển chuyển đến mệnh lệnh tiếp theo ngay sau thân cấu trúc **if**

Cấu Trúc Điều Khiển – **if...else...**

► Cú pháp:

```
if (biểu thức logic)  
{  
    // biểu thức logic đúng (true) thực hiện thân if  
    ... ..  
}  
else  
{  
    // biểu thức logic sai (false) thực hiện thân else  
    ... ..  
}  
... ..
```

Cấu Trúc Điều Khiển – **if** / **if...else...**

- ▶ Ví dụ: tìm giá trị tuyệt đối

```
int absVal;  
if (x < 0)  
{  
    absVal = -x;  
}  
else  
{  
    absVal = x;  
}
```

```
int absVal = x;  
if (x < 0)  
{  
    absVal = -x;  
}
```

Cấu Trúc Điều Khiển – **if** / **if...else...**

- ▶ Ví dụ: tìm số lớn, số bé

```
int min, max;  
if (first_num < second_num) {  
    min = first_num;  
    max = second_num;  
}  
else {  
    min = second_num;  
    max = first_num;  
}
```

```
int min = second_num, max = first_num;  
if (first_num < second_num) {  
    min = first_num;  
    max = second_num;  
}
```


Cấu Trúc Điều Khiển – **if** / **if...else...**

- ▶ Ví dụ: đổi số sang chữ

```
int number; cin >> number;
string number_to_word;
if (number == 0) number_to_word = "zero";
if (number == 1) number_to_word = "one";
if (number == 2) number_to_word = "two";
if (number == 3) number_to_word = "three";
if (number == 4) number_to_word = "four";
if (number == 5) number_to_word = "five";
if (number == 6) number_to_word = "six";
if (number == 7) number_to_word = "seven";
if (number == 8) number_to_word = "eight";
if (number == 9) number_to_word = "nine";
if (... .. <none of the above> ... ..)
    number_to_word = "I dont know";
cout << number_to_word << endl;
```

Cấu Trúc Điều Khiển – **if / if...else...**

- ▶ Ví dụ: đổi số sang chữ

```
int number; cin >> number;
string number_to_word;
if (number == 0) number_to_word = "zero";
else if (number == 1) number_to_word = "one";
else if (number == 2) number_to_word = "two";
else if (number == 3) number_to_word = "three";
else if (number == 4) number_to_word = "four";
else if (number == 5) number_to_word = "five";
else if (number == 6) number_to_word = "six";
else if (number == 7) number_to_word = "seven";
else if (number == 8) number_to_word = "eight";
else if (number == 9) number_to_word = "nine";
else /* if (... <none of the above> ...) */
    number_to_word = "I do not know";
cout << number_to_word << endl;
```

Cấu Trúc Điều Khiển – **if** / **if...else...**

- ▶ Ví dụ: xác định chữ cái thường, chữ cái hoa, chữ số, nếu không phải in ra thông báo

```
char ky_tu;
if (  ☐  ) {
    cout << "chu cai thuong" << endl;
}
if (  ☐  ) {
    cout << "chu cai hoa" << endl;
}
if (  ☐  ) {
    cout << "chu so" << endl;
}
if ( ...  ☐  ... ) {
    cout << "khong biet" << endl;
}
```

Cấu Trúc Điều Khiển – **if** / **if...else...**

- ▶ Ví dụ: xác định chữ cái thường, chữ cái hoa, chữ số, nếu không phải in ra thông báo

```
char ky_tu;
if ('a' <= ky_tu && ky_tu <= 'z') {
    cout << "chu cai thuong" << endl;
}
else if ('A' <= ky_tu && ky_tu <= 'Z') {
    cout << "chu cai hoa" << endl;
}
else if ('0' <= ky_tu && ky_tu <= '9') {
    cout << "chu so" << endl;
}
else /* if (... <= ky_tu && ky_tu <= ...) */ {
    cout << "khong biet" << endl;
}
```

Cấu Trúc Điều Khiển – **if** / **if...else...**

- ▶ Ví dụ: xếp loại A, B, C, D, E, F dựa trên điểm tổng kết

```
char rank;  
if (m_total >= 8.0)  
    rank = 'A';  
if (m_total >= 7.0 && m_total <x 8.0)  
    rank = 'B';  
if (m_total >= 6.0 && m_total <x 7.0)  
    rank = 'C';  
if (m_total >= 5.0 && m_total <x 6.0)  
    rank = 'D';  
if (m_total >= 4.0 && m_total <x 5.0)  
    rank = 'E';  
if (m_total <x 4.0)  
    rank = 'F';  
cout << "xep loai: " << rank << endl;
```

Cấu Trúc Điều Khiển – **if / if...else...**

- ▶ Ví dụ: xếp loại A, B, C, D, E, F dựa trên điểm tổng kết

```
char rank;  
if (m_total >= 8.0)  
    rank = 'A';  
else if (m_total >= 7.0) /* && m_total < 8.0 */  
    rank = 'B';  
else if (m_total >= 6.0) /* && m_total < 7.0 */  
    rank = 'C';  
else if (m_total >= 5.0) /* && m_total < 6.0 */  
    rank = 'D';  
else if (m_total >= 4.0) /* && m_total < 5.0 */  
    rank = 'E';  
else  
    rank = 'F';  
cout << "xep loai: " << rank << endl;
```

Cấu Trúc Điều Khiển – **if** / **if...else...**

- ▶ Ví dụ: xếp loại A, B, C, D, E, F dựa trên điểm tổng kết

```
char rank;  
if (m_total >= 4.0)  
    rank = 'E';  
else if (m_total >= 5.0)  
    rank = 'D';  
else if (m_total >= 6.0)  
    rank = 'C';  
else if (m_total >= 7.0)  
    rank = 'B';  
else if (m_total >= 8.0)  
    rank = 'A';  
else  
    rank = 'F';  
cout << "xep loai: " << rank << endl;
```

SAI !!!

rank
chỉ 'E'
hoặc 'F'

Cấu Trúc Điều Khiển – **if** / **if...else...**

```
char rank = 'F';  
if (m_total >= 8.0)  
    rank = 'A';  
    count_A = count_A + 1;  
else  
    ... ..
```

- ▶ Lỗi thường xảy ra:

error: 'else' without a previous if

Cấu Trúc Điều Khiển – **if** / **if...else...**

```
if (x > 5)
    if (y > 5)
        cout << "x va y > 5";
else
    cout << "x <= 5";
```

▶ **else** của **if** nào?

- ▶ Khi **x = 5**, mệnh lệnh nào thực hiện, in ra?
- ▶ **else** của **if** gần nhất

Phép Toán So Sánh

Phép Toán	Toán Tử	Ví Dụ	Kết Quả
Nhỏ hơn	<	2.0 < 1.0 1 < 2	false true
Nhỏ hơn hoặc bằng	<=	2.0 <= 1.0 1 <= 2	false true
Lớn hơn	>	2.0 > 1.0 1 > 2	true false
Lớn hơn hoặc bằng	>=	1.0 >= 1.0 1 >= 2	true false
Bằng	==	1 == 1 1 == 2	true false
Không bằng (khác)	!=	2 != 2 1 != 2	false true

Phép Toán Logic

Phép Toán	Toán Tử	Ví Dụ	Kết Quả
Và	&&	true && false true && true	false true
Hoặc	 	true false false false	true false
Phủ định	!	!false !true	true false

Bảng Logic

a	b	a && b
true	true	true
true	false	false
false	true	false
false	false	false

a	b	a b
false	false	false
false	true	true
true	false	true
true	true	true

```
int a = 0;  
bool b;  
b = (false) && (1/a); cout << b;  
b = (true) || (1/a); cout << b;
```

Độ Ưu Tiên Các Phép Toán

- Xác định thứ tự để tính giá trị biểu thức

Độ Ưu Tiên	Toán Tử
Cao nhất	$++$, $--$, $!$
	$*$, $/$, $\%$
	$+$, $-$
	$<$, $<=$, $>$, $>=$
	$==$, $!=$
	$\&\&$
	$ $
Thấp nhất	$=$, $+=$, $-=$, $*=$, $/=$, $\%=$

$x + 1 > 2 \quad || \quad x + 1 < -3$ tương đương:
 $((x + 1) > 2) \quad || \quad ((x + 1) < -3)$

Biểu Thức Logic

- ▶ Các phép toán logic
 - ▶ PHỦ ĐỊNH (**!**), VÀ (**&&**), HOẶC (**||**)
 - ▶ mức độ ưu tiên: **&&** cao hơn **||**
- ▶ Kiểu dữ liệu cơ bản logic **bool**
 - ▶ sai (**false**), đúng (**true**)
 - ▶ chuyển đổi kiểu dữ liệu:
 - không **0** hoặc **0.0** là **false**
 - khác không là **true**
 - **false** là **0**
 - **true** là **1**

Biểu Thức Logic Kết Hợp

- ▶ Biểu thức logic trong toán học

$(0.0 \leq m_total < 4.0)$

- ▶ Chuyển trực tiếp sang ngôn ngữ lập trình

$(0.0 \leq m_total < 4.0)$

dịch không lỗi nhưng chạy lỗi, **sai !!!**

- ▶ Biểu thức logic trong ngôn ngữ lập trình

- ▶ phải sử dụng các phép toán logic

$(0.0 \leq m_total \ \&\& \ m_total < 4.0)$

- ▶ lỗi thường gặp: biểu thức logic luôn sai/đúng

$(0.0 \geq m_total \ \&\& \ m_total > 4.0)$

Một Số Lưu Ý

- ▶ Nhầm lẫn giữa gán (=) và so sánh bằng (==)

- ▶ có thể không lỗi khi dịch

- ▶ kết quả chạy khác nhau

- `if (a = 1)` // điều kiện luôn đúng

- `if (a = 0)` // điều kiện luôn sai

- `if (a = b)` // tương đương

- `if ((a = b) != 0)`

- ▶ thực hiện phép gán xong rồi kiểm tra điều kiện

- không 0 hoặc 0.0 là `false` / `false` là 0

- khác không là `true` / `true` là 1

- ▶ dịch báo lỗi: `if (a*a = b*b + c*c)`

- lỗi cú pháp phép gán (bên trái là biến số)

Cấu Trúc Điều Khiển – **if** / **if...else...**

- ▶ Ví dụ: đổi số sang chữ

```
int num; cin >> num;
string number_to_word;

if (num == 0) number_to_word = "zero";
if (num == 1) number_to_word = "one";
if (num == 2) number_to_word = "two";
if (num == 3) number_to_word = "three";
if (num == 4) number_to_word = "four";
if (num == 5) number_to_word = "five";
if (num == 6) number_to_word = "six";
if (num == 7) number_to_word = "seven";
if (num == 8) number_to_word = "eight";
if (num == 9) number_to_word = "nine";
if (... ..) number_to_word = "I dont know";
```

Cấu Trúc Điều Khiển – **switch**

► Ví dụ: đổi số sang chữ

```
int num; cin >> num;
string number_to_word;
switch (num) {
    case 0: number_to_word = "zero"; break;
    case 1: number_to_word = "one"; break;
    case 2: number_to_word = "two"; break;
    case 3: number_to_word = "three"; break;
    case 4: number_to_word = "four"; break;
    case 5: number_to_word = "five"; break;
    case 6: number_to_word = "six"; break;
    case 7: number_to_word = "seven"; break;
    case 8: number_to_word = "eight"; break;
    case 9: number_to_word = "nine"; break;
    default: number_to_word = "I dont know";
}
```

Cấu Trúc Điều Khiển – **if** / **if...else...**

- ▶ Ví dụ: in lịch sinh hoạt, làm việc trong tuần

```
int day; string act;

if (day == 2) act = "hoc chieu";
if (day == 3) act = "hoc sang";
if (day == 4) act = "di lam";
if (day == 5) act = "hoc sang va chieu";
if (day == 6) act = "hoc chieu va toi";
if (day == 7) act = "choi the thao";
if (day == 8) act = "di sang nha ban";
if (... ..) act = "o nha ngu";
```

Cấu Trúc Điều Khiển – **switch**

- ▶ Ví dụ: in lịch sinh hoạt, làm việc trong tuần

```
int day; string act;
switch (day)
{
    case 2: act = "hoc chieu"; break;
    case 3: act = "hoc sang"; break;
    case 4: act = "di lam"; break;
    case 5: act = "hoc sang va chieu"; break;
    case 6: act = "hoc chieu va toi"; break;
    case 7: act = "choi the thao"; break;
    case 8: act = "di sang nha ban"; break;
    default: act = "o nha ngu";
}
```

Cấu Trúc Điều Khiển – **switch**

- ▶ Nếu không có **break** sẽ thực hiện các **case** tiếp theo đến khi nào gặp **break** thì thoát khỏi **switch**

```
int day; string act;
switch (day)
{
    case 2:
    case 3:
    case 5:
    case 6: act = "đi học"; break;
    case 4: act = "đi làm"; break;
    case 7:
    case 8: act = "đi chơi"; break;
    default: act = "o nha ngu";
}
```

Cấu Trúc Điều Khiển – **switch**

- ▶ Ví dụ: tính số ngày trong tháng dương lịch

```
int year, month, num_day;
cin >> month;
cin >> year;
switch (month)
{
    case 2:
        num_day = ... ..; break;
    case 4: case 6: case 9: case 11:
        num_day = 30; break;
    default:
        num_day = 31; break;
}
... ..
```

Cấu Trúc Điều Khiển – **switch**

```
switch (selection)
{
    case 1: cout << "cash withdrawal"; break;
    case 2: cout << "change PIN"; break;
    case 3: cout << "transfer fund"; break;
    case 4: cout << "balance inquiry"; break;
    default: cout << "return card";
}
```

```
switch (selection)
{
    case 12: cout << "choose CocaCola"; break;
    case 23: cout << "choose Revive"; break;
    case 34: cout << "choose Milk"; break;
    case 45: cout << "choose Lavie"; break;
    default: cout << "clean the kitchen";
}
```

Cấu Trúc Điều Khiển – **switch**

► Cú pháp:

```
switch (Biểu Thức)
```

```
{
```

```
    case HàngSố1: CụmMệnhLệnh1; break;
```

```
    case HàngSố2: CụmMệnhLệnh2; break;
```

```
    case HàngSố3: CụmMệnhLệnh3; break;
```

```
    ... ..
```

```
    default: CụmMệnhLệnhMặcĐịnh;
```

```
}
```

- Lưu ý: kiểu trả về của **Biểu Thức** và kiểu của **HàngSố** của nhãn **case** phải là kiểu số nguyên (**int**, **short**, **long**, **char**, **bool**)

Cấu Trúc Điều Khiển – **switch**

- ▶ Cú pháp:

```
switch (Biểu Thức)  
{  
    case HằngSố: CụmMệnhLệnh; break;  
}
```

- ▶ Ý nghĩa:

- ▶ Nhãn **case** được xét tuần tự
- ▶ Giá trị **HằngSố** của **case** nào ứng với giá trị của **Biểu Thức**, **CụmMệnhLệnh** của **case** đó thực hiện
- ▶ Khi gặp **break**, thì thoát khỏi cấu trúc **switch**
- ▶ Thực hiện **CụmMệnhLệnhMặcĐịnh** nếu không có **case** nào tương ứng

Bài Tập

- ▶ Bài tập (xem mã nguồn)
 1. Kiểm tra số nhập vào chẵn hay lẻ
 2. Tìm số lớn nhất trong 3 số nhập vào
 3. Đoán số