



THỰC HỌC – THỰC NGHIỆP



Conceive Design Implement Operate

BÀI 3: LỆNH RẼ NHÁNH

ĐẠI CƯƠNG LẬP TRÌNH



- Biết được cách vẽ lưu đồ thuật toán
- Biết cách sử dụng lệnh if-else
- Biết cách sử dụng lệnh switch-case

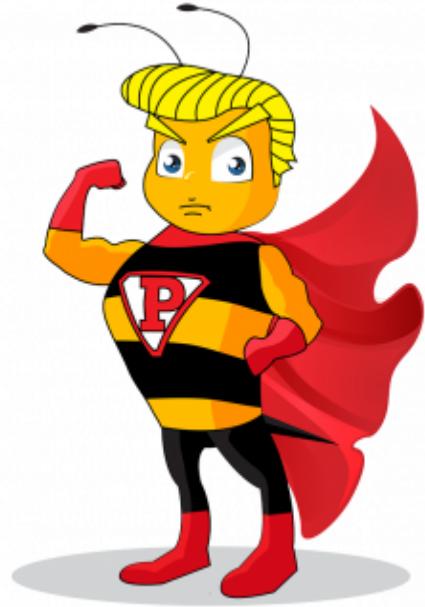
Khái niệm về lưu đồ thuật toán

Khái niệm về rẽ nhánh trong lập trình

Giới thiệu về lệnh if-else

Giới thiệu về lệnh switch-case





PHẦN 1: LỆNH RẼ NHÁNH - IF

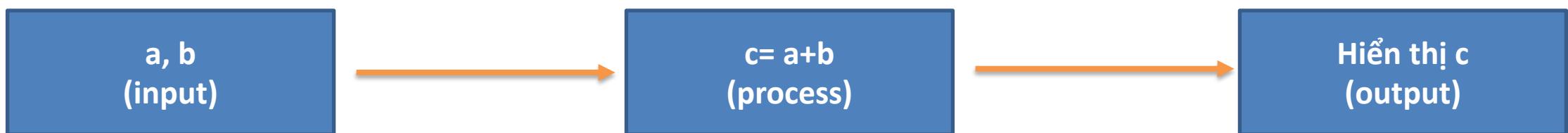


PHÂN TÍCH BÀI TOÁN

- ❑ Để giải quyết một bài toán bằng lập trình, chúng ta cần phải phân tích kỹ và xác định các bước



- ❑ Ví dụ: xây dựng chương trình tính tổng hai số



- ❑ Sau khi đã phân tích bài toán, chúng ta cần biểu diễn cách giải bài toán dưới 2 dạng
 - ❖ Ngôn ngữ tự nhiên (*natural languages*)
 - ❖ Mã giả (*pseudocode*)
 - ❖ Lưu đồ thuật toán (*flowcharts*)

Ví dụ: Xây dựng chương trình chia 2 số bất kỳ (a/b)

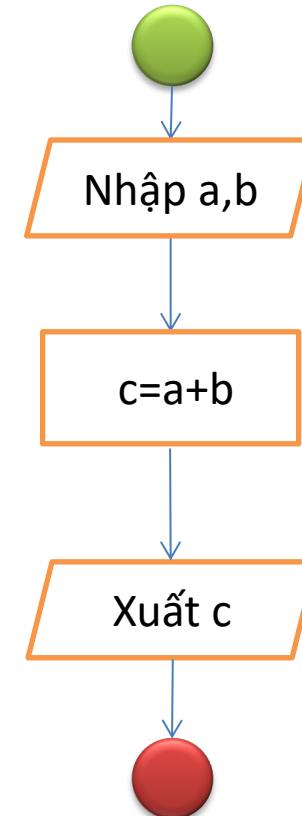
- ❖ Bước 1: Nhập 2 số a,b
- ❖ Bước 2: Nếu $b = 0$, hiển thị vô nghiệm
- ❖ Bước 3: Ngược lại, $c = a/b$
- ❖ Bước 4: Hiển thị c, kết thúc chương trình

LƯU ĐỒ THUẬT TOÁN (FLOWCHART)

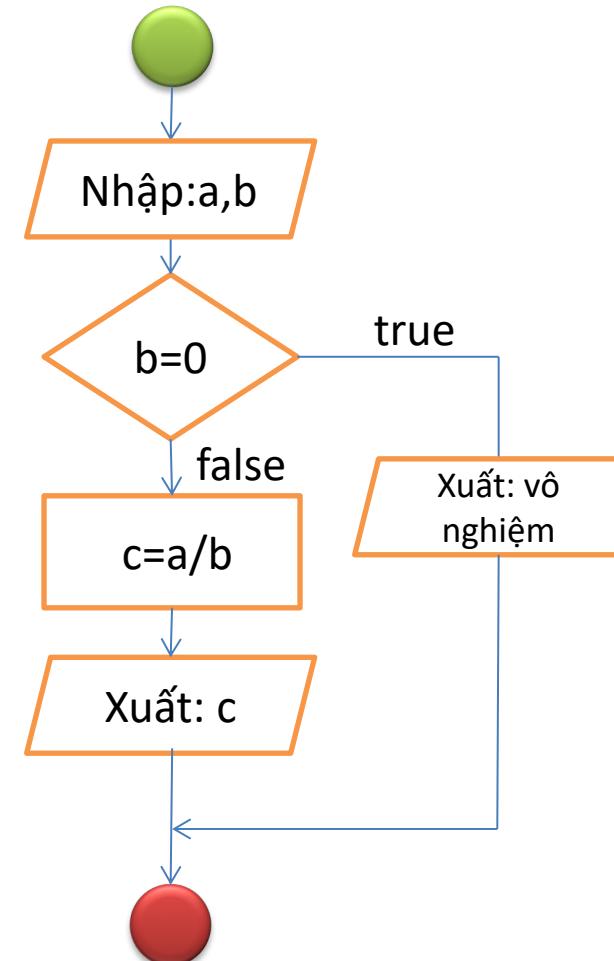
- ❑ Lưu đồ thuật toán là công cụ dùng để biểu diễn thuật toán, mô tả nhập (input), dữ liệu xuất (output) và luồng xử lý qua các ký hiệu hình học

STT	Ký hiệu	Ý nghĩa
1		Bắt đầu/ kết thúc
2		Điều kiện rẽ nhánh (lựa chọn)
3		Nhập hoặc xuất
4		Tính toán
5		Luồng xử lý

- ❑ Ví dụ: Chương trình cộng 2 số bất kỳ a+b



- ❑ Ví dụ: Chương trình chia 2 số bất kỳ (a/b)





PHẦN 1: LỆNH RẼ NHÁNH - IF

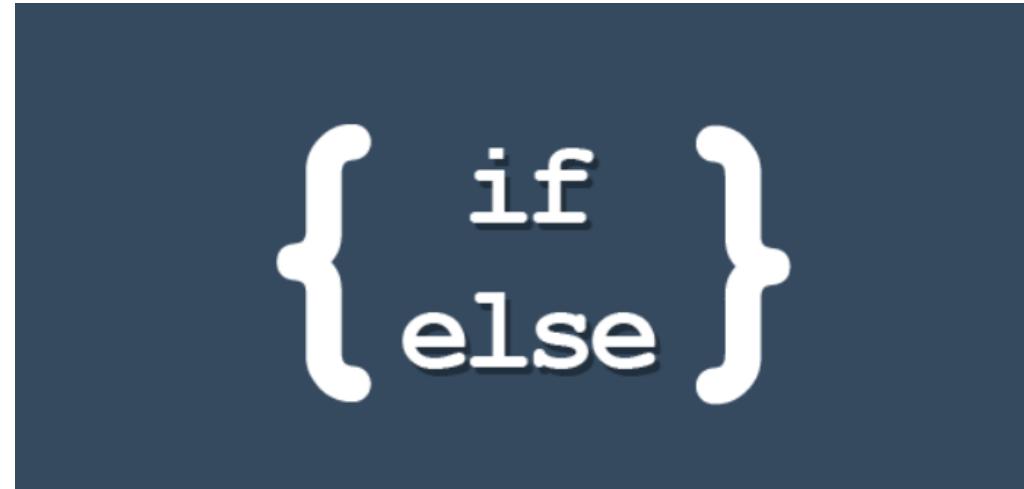
Bạn có bao giờ phải đau đầu trong việc lựa chọn?

- ❖ Nếu trời mưa thì ở nhà hay đi học
- ❖ Nếu có thời gian rảnh vào cuối tuần thì đi xem phim hay đi xem bóng đá
- ❖ Nếu hôm nay thầy cho nghỉ học thì ở nhà học bài hay là cà chơi game

Nếu ... thì là những tình huống chúng ta rất thường xuyên gặp trong cuộc sống hằng ngày



- ❑ Cấu trúc rẽ nhánh trong mỗi ngôn ngữ lập trình luôn luôn là một thành phần quan trọng.
- ❑ Đó là thành phần cơ bản tạo nên hầu hết các thuật toán hiện đại.
- ❑ Bạn muốn mô tả một đoạn chương trình kiểu như: Nếu gặp điều kiện C thì thực hiện hành động A, ngược lại, thực hiện hành động B thì bạn chắc chắn cần sử dụng cấu trúc rẽ nhánh.



{ if
else }

❑ Lệnh if (rẽ nhánh) được sử dụng để ra quyết định thực hiện một công việc nào đó tùy vào điều kiện đúng hay sai.

❑ Có 3 hình thức rẽ nhánh

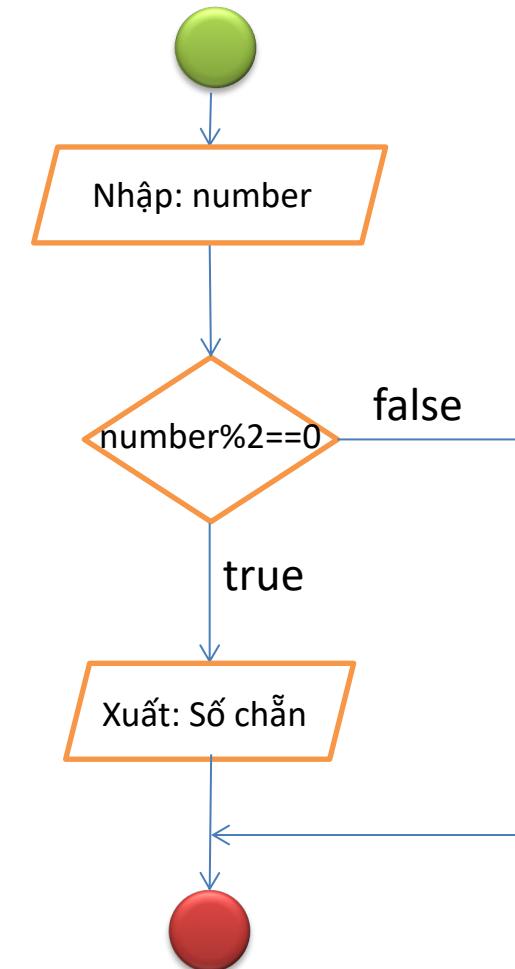
- ❖ if(điều kiện){...}
- ❖ if(điều kiện){...}else{...}
- ❖ if(điều kiện 1){...} ...else if(điều kiện i){...}...else{...}

❑ Ví dụ

```

if(number % 2 == 0){
    printf("Số chẵn\n");
}

```

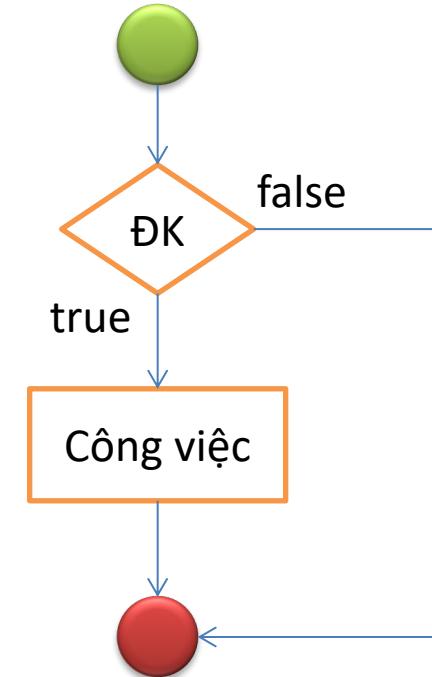


Cú pháp

```
if(<<điều kiện>>) {  
    << Công việc >>  
}
```

 Diễn giải:

- ❖ Nếu **điều kiện** có giá trị true thì **công việc** được thực hiện



❑ Ví dụ:

```
double diem = 4;  
if (diem >= 5) {  
    printf("Đậu");  
}
```

❑ Diễn giải:

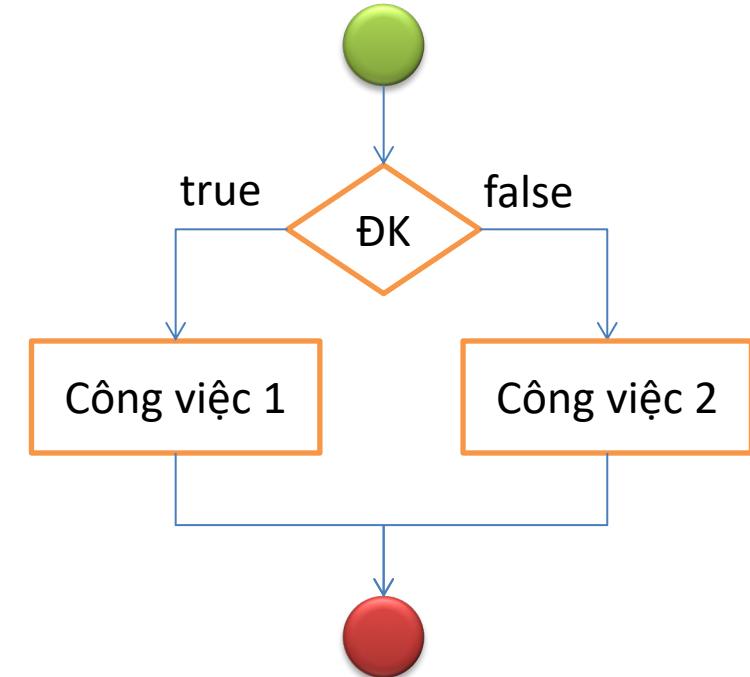
- ❖ Đoạn mã trên không xuất gì ra màn hình cả vì biểu thức điều kiện **diem >= 5** có giá trị false

❑ Cú pháp

```
if (<<điều kiện>>) {  
    << công việc 1 >>  
}  
  
else {  
    << công việc 2 >>  
}
```

❑ Diễn giải

- ❖ Nếu **điều kiện** có giá trị true thì **công việc 1** được thực hiện, ngược lại **công việc 2** được thực hiện

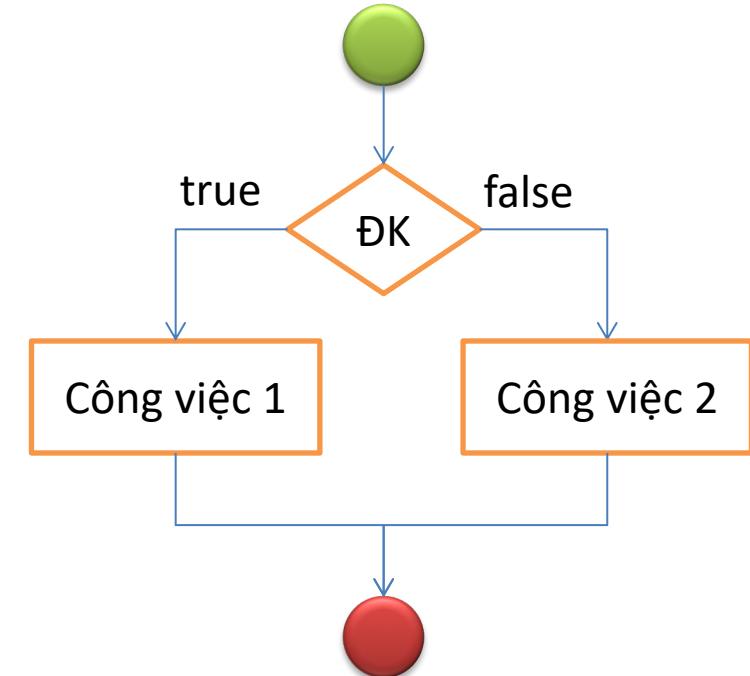


Ví dụ

```
double diem = 4;  
if (diem < 5) {  
    printf("Rớt\n");  
}  
else {  
    printf("Đậu\n");  
}
```

 Diễn giải:

- ❖ Đoạn mã trên xuất chữ “Rớt” ra màn hình vì điều kiện **diem < 5** có giá trị là **true**.

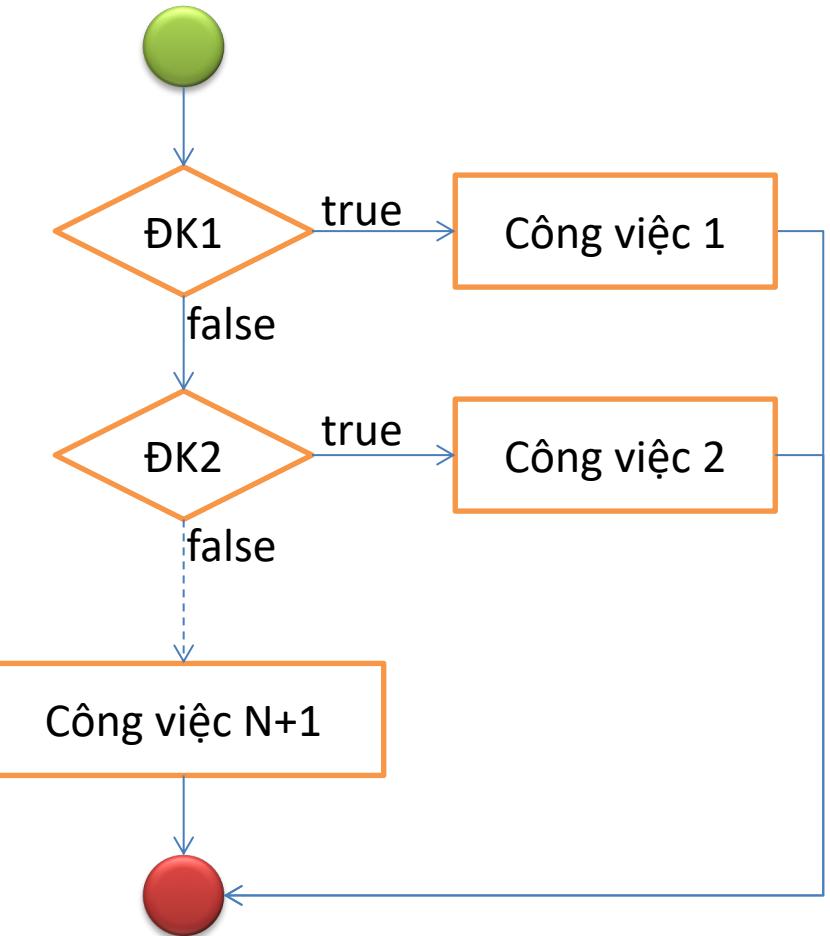


Cú pháp

```
if (<<điều kiện 1>>){  
    << công việc 1 >>  
}  
else if (<<điều kiện 2>>){  
    << công việc 2 >>  
}  
...  
else {  
    << công việc N+1 >>  
}
```

 Diễn giải

- ❖ Chương trình sẽ kiểm tra từ **điều kiện 1 đến N** nếu gặp **điều kiện i** đầu tiên có giá trị true thì sẽ thực hiện **công việc i**, ngược lại sẽ thực hiện **công việc N+1**



❑ Ví dụ: Xây dựng chương trình tính học lực của sinh viên theo điểm trung bình

❖ Input: nhập vào điểm của sinh viên

❖ Process:

➢ Nếu điểm ≥ 9 : xuất sắc

➢ Nếu $9 >$ điểm ≥ 8 : giỏi

➢ Nếu $8 >$ điểm ≥ 7 : khá

➢ Nếu $7 >$ điểm ≥ 5 : trung bình

➢ Nếu $5 >$ điểm ≥ 3 : yếu

➢ Ngược lại: kém

❖ Output: hiển thị học lực của sinh viên



LỆNH RẼ NHÁNH IF

- Lưu đồ thuật toán
- Khái niệm lệnh rẽ nhánh
- Lệnh if-else





CHÈN QUIZ



PHẦN 2: LỆNH RẼ NHÁNH SWITCH

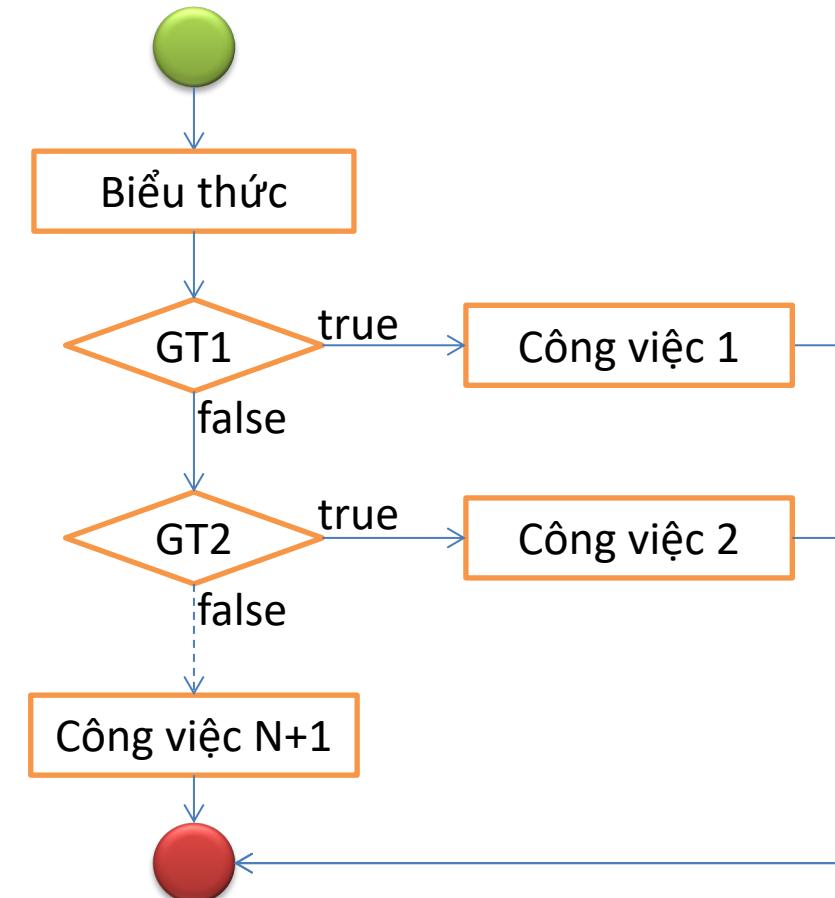
- ❑ Lệnh switch-case cũng tương tự như lệnh if-else-if mà chúng ta đã được học ở bài trước. Nghĩa là nó có nhiều điều kiện, chương trình chúng ta duyệt từng điều kiện từ trên xuống dưới, nếu thỏa điều kiện nào thì đoạn code bên trong điều kiện đó sẽ được thực thi.
- ❑ Lệnh switch cũng giống cấu trúc if, nhưng nó mềm dẻo hơn và linh động hơn nhiều so với sử dụng if.
- ❑ Tuy nhiên, nó cũng có mặt hạn chế là kết quả của biểu thức phải là giá trị hằng nguyên (có giá trị cụ thể)

❑ Cú pháp

```
switch (<<biểu thức>>)
{
    case <<giá trị số>>:
        // Công việc 1
        break;
    case <<giá trị sô>>:
        // Công việc 2
        break;
    ...
    default:
        // Công việc N+1
        break;
}
```

❑ Diễn giải

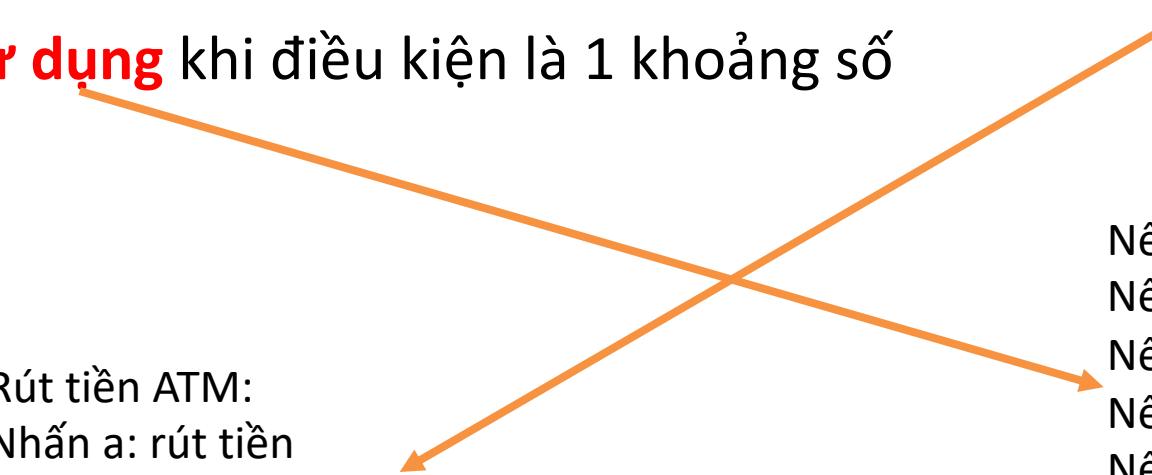
- ❖ Nếu <biểu thức> = <giá trị i> thì thực hiện <công việc i> ngược lại sẽ thực hiện <công việc n+1>.
- ❖ Nếu case không chứa break thì case tiếp sau sẽ được thực hiện



☐ Khi nào sử dụng switch – case

- ❖ Chỉ sử dụng khi điều kiện cần kiểm tra là một '**số nguyên**'
- ❖ Thường được sử dụng để xây dựng **menu cho chương trình**
- ❖ **Không thể sử dụng** khi điều kiện là 1 khoảng số

Rút tiền ATM:
Nhấn a: rút tiền
Nhấn b: xem số dư
Nhấn c: chuyển khoản
Nhấn d: thoát chương trình



Nếu điểm ≥ 9 : xuất sắc
Nếu $9 >$ điểm ≥ 8 : giỏi
Nếu $8 >$ điểm ≥ 7 : khá
Nếu $7 >$ điểm ≥ 5 : trung bình
Nếu $5 >$ điểm ≥ 3 : yếu
Ngược lại: kém

❑ Ví dụ: Xây dựng chương trình trả lời câu hỏi trắc nghiệm.

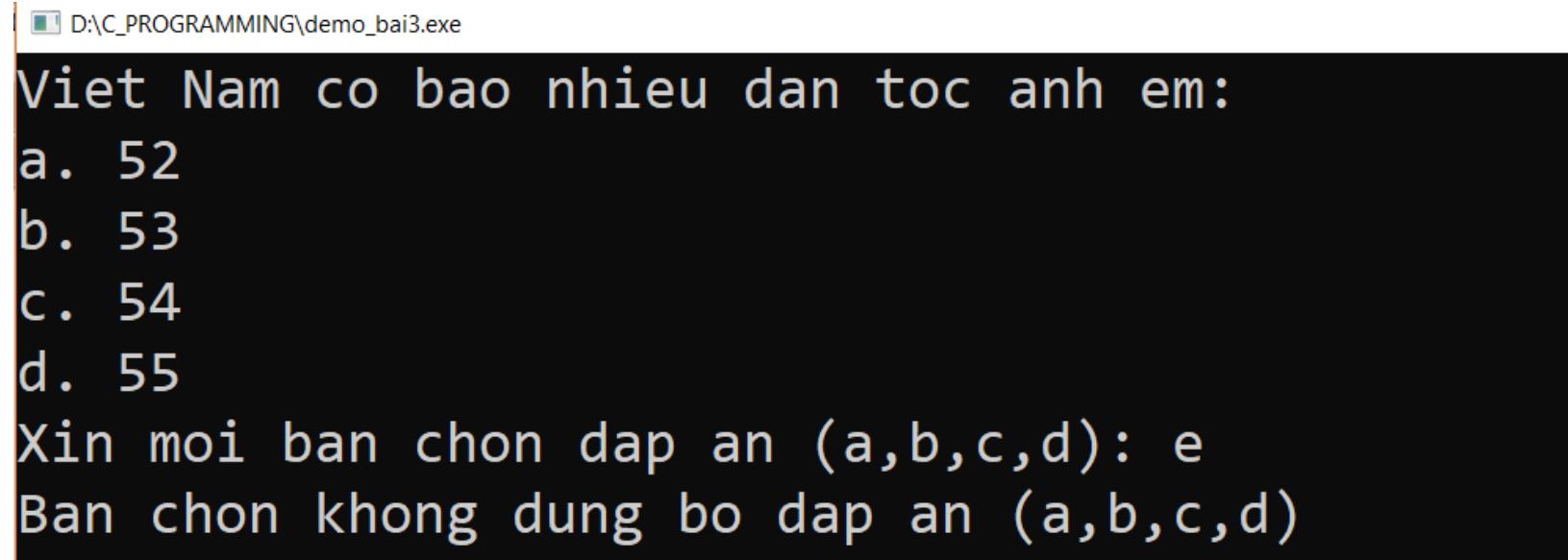
- ❖ Hiển thị câu hỏi: Việt Nam có bao nhiêu dân tộc, xin mời người dùng lựa chọn đáp án (a,b,c,d)

```
D:\C_PROGRAMMING\demo_bai3.exe
Viet Nam co bao nhieu dan toc anh em:
a. 52
b. 53
c. 54
d. 55
Xin moi ban chon dap an (a,b,c,d):
```

```
printf("Viet Nam co bao nhieu dan toc anh em:\n");
printf("a. 52\nb. 53\nc. 54\nd. 55\n");
printf("Xin moi ban chon dap an (a,b,c,d): ");
char dapAn;
scanf("%s",&dapAn);
switch(dapAn){
    case 'a': printf("Ban da chon sai\n");
    break;
    case 'b': printf("Ban da chon sai\n");
    break;
    case 'c': printf("Ban da chon dung\n");
    break;
    case 'd': printf("Ban da chon sai\n");
    break;
}
}
```

❑ Ví dụ: Xây dựng chương trình trả lời câu hỏi trắc nghiệm.

- ❖ Hiển thị câu hỏi: Việt Nam có bao nhiêu dân tộc, xin mời người dùng lựa chọn đáp án (a,b,c,d)
- ❖ Hiển thị: Bạn đã chọn không đúng bộ đáp án khi người dùng nhập các chữ cái khác (a,b,c,d) → Thêm default trong cấu trúc switch – case



```
D:\C_PROGRAMMING\demo_bai3.exe
Viet Nam co bao nhieu dan toc anh em:
a. 52
b. 53
c. 54
d. 55
Xin moi ban chon dap an (a,b,c,d): e
Ban chon khong dung bo dap an (a,b,c,d)
```

```
printf("Viet Nam co bao nhieu dan toc anh em:\n");
printf("a. 52\nb. 53\nc. 54\nd. 55\n");
printf("Xin moi ban chon dap an (a,b,c,d): ");
char dapAn;
scanf("%s",&dapAn);
switch(dapAn){
    case 'a': printf("Ban da chon sai\n");
    break;
    case 'b': printf("Ban da chon sai\n");
    break;
    case 'c': printf("Ban da chon dung\n");
    break;
    case 'd': printf("Ban da chon sai\n");
    break;
    default:
        printf("Ban chon khong dung bo dap an (a,b,c,d)");
```



LẼNH SWITCH

- Giai đoạn 1: Xây dựng menu chương trình
- Giai đoạn 2: Xây dựng chức năng trong chương trình
- Final: Hoàn thiện chương trình

- 1** Chức năng 1
- 2** Chức năng 2
- 3** Chức năng 3

- Nội dung chức năng 1
- Nội dung chức năng 2
- Nội dung chức năng 3

Giai đoạn 1: Xây dựng menu chương trình

Diễn giải:

- ❖ Chương trình có 10 chức năng riêng biệt.
- ❖ Khi chạy, chương trình sẽ hiển thị một menu để người dùng lựa chọn
- ❖ Người dùng sẽ chọn bằng cách bấm vào các số từ 1-10



HƯỚNG DẪN ASSIGNMENT GĐ 1

- Lệnh switch-case
- Ứng dụng lệnh switch-case để làm menu





CHÈN QUIZ



thank
you!