

JAVA**CẤU TRÚC LẬP TRÌNH****1. Hệ số nhị phân và hệ số HEX, cơ chế xử lý thông tin trên máy tính**

Trong máy tính thông tin được biểu diễn bằng bit 0 và 1 người ta quy định là Hệ số nhị phân được tạo ra từ các số 0, 1 này.

1 byte = 8 bit (bit có giá trị là 0 hoặc 1)

- **Chuyển từ số thập phân sang số nhị phân:** Bằng cách lấy số đó chia cho 2 được thương tiếp tục chia cho 2 đến khi thương bằng 0. Mỗi lần chia cần ghi số dư. Sau đó khi thương bằng 0 sẽ viết ngược lại thứ tự số dư từ cuối lên đầu thì được số cần tìm.

Ví dụ:

Cần chuyển số thập phân sang nhị phân

Số 12 => số nhị phân

$$12/2 = 6 \text{ dư } 0$$

$$6/2 = 3 \text{ dư } 0$$

$$3/2 = 1 \text{ dư } 1$$

$$1/2 = 0 \text{ dư } 1$$

=> Đảo chuỗi sắp xếp từ dưới lên: 1100

$$12 \Rightarrow 1100$$

Vậy số nhị phân cần tìm của số 12 là: 1100

Ngược lại ta cũng chuyển từ số nhị phân về số thập phân bằng cách lấy giá trị 2^n như sau:

$$1100 \Rightarrow 1 \times 2^3 + 1 \times 2^2 + 0 + 0 = 8 + 4 = 12$$

- **Hệ số thập lục (HEX):** Là hệ số được sử dụng để biểu diễn gọn gàng cho hệ số nhị phân thường được sử dụng để quy định về mã màu trong đồ họa, lập trình web,...

Quy định giá trị từ 0 đến 15 trong đó:

10=> A

11=> B

12=> C

13=> D

14=> E

15=>F

Ví dụ: #06AAD6 chuyển về hệ số nhị phân là: 0000.0101.1010.1010.1101.0101

2. Kiểu dữ liệu, toán tử trong lập trình Java

Các kiểu dữ liệu trong ngôn ngữ lập trình Java như kiểu int, float, long, string,... khác nhau ở kích thước cấp phát vùng nhớ để lưu trữ thông tin trong quá trình xử lý khi chạy chương trình trên máy tính.

Kích thước kiểu dữ liệu được tính bằng **byte**, từ đó có thể đưa ra được vùng giá trị của kiểu dữ liệu tương ứng bằng cách quy đổi từ byte ra bit sau đó vùng giá trị sẽ là: 2^n trong đó n là số bit được quy đổi.

- **Toán tử trong lập trình Java:** được sử dụng để thực hiện các phép tính toán hoặc so sánh thông tin các biến, biểu thức với nhau.

Ví dụ:

int x=10; int y=5; //Khai báo biến kiểu số nguyên và gán giá trị ban đầu cho biến

x++; //Tăng giá trị của biến lên 1 đơn vị

x--; //Giảm giá trị của biến 1 đơn vị

x+=y => x= x+y = 10 + 5 = 15

x-=y => x=x-y = 10-5 = 5;

x*=y => x= x*y = 10 * 5 = 50

x/=y => x= x/y = 10/5 = 2;

x%=y => x = x%y = 10%5 = 0;

Chia lấy dư: $3\%10 = 3$

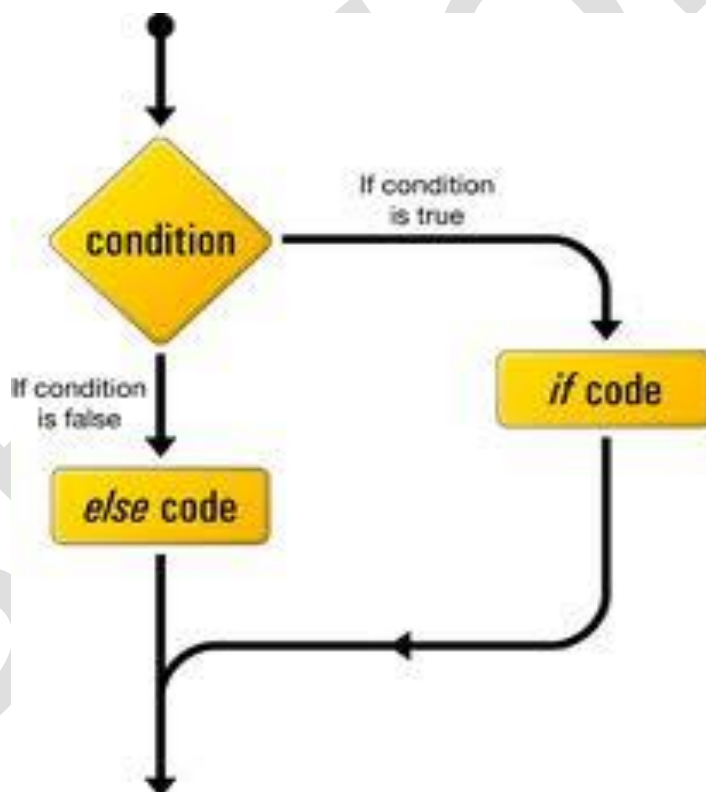
Phép chia: $3/10 = 0$ nếu biến là dạng số nguyên

3. Cấu trúc lập trình trong Java

Để yêu cầu xử lý các công việc trên máy tính, người lập trình cần viết các đoạn mã để yêu cầu máy tính xử lý theo các cú pháp dựa trên các từ khóa quy định trong ngôn ngữ Java.

- **Cấu trúc If...Else:** Chương trình sẽ thực hiện một hoặc nhiều công việc được viết trong khối `{ }` của từ khóa **If** nếu biểu thức điều kiện đúng. Ngược lại sẽ thực hiện xử lý các công việc trong khối `{ }` của từ khóa **Else**.

Nếu thỏa mãn điều kiện này thì thực hiện công việc này ngược lại sẽ thực hiện công việc kia.



if(biểu thức điều kiện)

{

```
//Nếu có nhiều công việc cần xử lý  
}  
else  
{  
    //Sẽ thực hiện các công việc trong đây nếu  
    //biểu thức điều kiện không đúng hoặc sai  
}
```

Hoặc viết dưới dạng đơn giản nếu chỉ thực hiện một công việc duy nhất cú pháp như sau:

if(biểu thức điều kiện)

//Viết code thực hiện công việc

Ví dụ:

//Khai báo một biến kiểu số nguyên trong java

```
int a=4; int b = 5;
```

```
int thuong =0;
```

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

```
{
```

```
    thuong = a/b;
```

```
}
```

//In kết quả ra màn hình DOS

```
System.out.print("Thuong cua 2 so la: " + thuong);
```

- **Ghi chú trong lập trình Java**

Sử dụng ký hiệu `//` để comment từng dòng code trong java

Sử dụng `/* Comment */` nếu muốn comment nhiều dòng code trong java để khi chạy không thực hiện đoạn mã đó nữa.

- **Các bước để biên dịch một chương trình java bằng Javac**

Bước 1:

- Mở giao diện Command (Windows + R và gõ cmd)
- Trỏ đến thư mục cần biên dịch và chạy chương trình viết bằng Java.

cd path_folder_java

path_folder_java: Là thư mục chứa file java cần biên dịch

Ví dụ: E:\Java\HCCG\Exercises-01

Bước 2: Trỏ đến thư mục cài JDK và có javac để biên dịch chương trình

path path_folder_javac

path_folder_javac: Là thư mục khi cài JDK lên chứa javac

Ví dụ: C:\Program Files\Java\jdk1.7.0_09\bin

- **Cú pháp khai báo và tạo một phương thức trong Java**

Phương thức là một tập các khối lệnh (code) được người lập trình viết để xử lý các công việc khi được gọi. Để tạo ra một phương thức trong lập trình java người ta sử dụng cú pháp như sau:

[Bổ từ truy cập] [Kiểu dữ liệu] Ten_phuong_thuc(par1, par2,...parn)

{

//Nội dung xử lý công việc của hàm (method)

}

Trong đó:

- **Bổ từ truy cập:** Phạm vi truy cập gồm:

- **private:** chỉ truy cập trong bản thân lớp
- **protected:** chỉ truy xuất trong bản thân lớp và lớp kế thừa từ lớp đó
- **public:** Có thể truy xuất ở bất kỳ lớp nào
- **Kiểu dữ liệu:** có thể trả về kiểu int, float, string,... và dùng từ khóa return để trả về giá trị đó. Hoặc không có dữ liệu trả về thì dùng từ khóa void.
- **par1, par2,...,parn:** là các tham số của phương thức nếu có

Ví dụ:

//Phương thức tính tổng 2 số

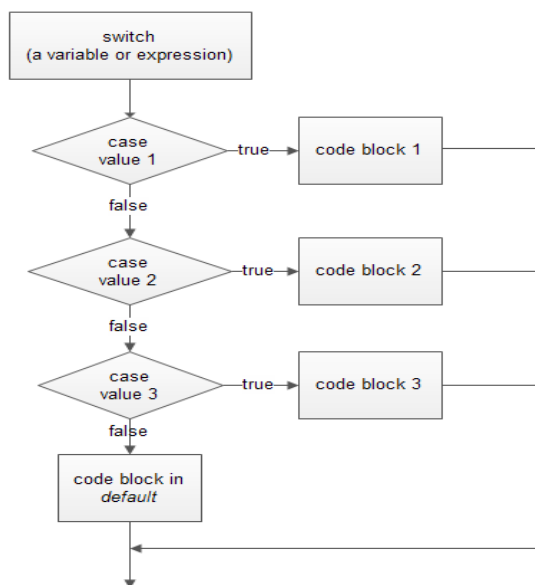
```
public static int TongSo(int a, int b)
{
    return a + b;
}
```

✓ Sử dụng Switch...Case trong Java

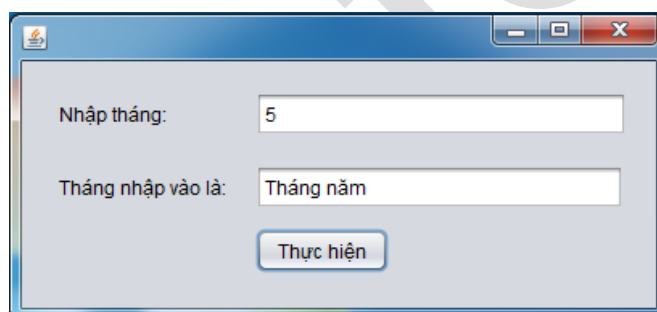
Là cấu trúc được sử dụng khi có nhiều giá trị cần so sánh thay vì sử dụng If...Else nhiều lần. Khi so khớp được giá trị sẽ thực hiện các công việc trong phần đó thôi sau đó sẽ thoát ra khỏi cấu trúc Switch.

Sử dụng từ khóa default để trả về giá trị mặc định trong trường hợp biểu thức so sánh không khớp với giá trị nào cần so sánh.

Cú pháp:



Ví dụ: Nhập tháng vào trên giao diện sau đó in thông tin tháng dạng chữ lên trên giao diện khi nhấn nút “**Thực hiện**” với giao diện được viết trong NetBeans như sau:



```

//Khai bao mot bien de lay thang ma nguoi dung nhap vao
int sothang=0;

//Khai bao mot bien chuoi de luu tru ket qua
String strKetQua = "";

//Lay thong tin thang nhap vao tu giao dien
//Chuyen ve kieu so nguyen, dong thoi gan cho bien sothang
sothang = Integer.parseInt(txtNhapThang.getText());

//Kiem tra thang nguoi dung nhap vao
switch(sothang)
  
```

```
{  
    case 1: strKetQua = "Tháng một";  
        break;  
    case 2: strKetQua = "Tháng hai";  
        break;  
    case 3: strKetQua = "Tháng ba";  
        break;  
    case 4: strKetQua = "Tháng tư";  
        break;  
    case 5: strKetQua = "Tháng năm";  
        break;  
    case 6: strKetQua = "Tháng sáu";  
        break;  
    case 7: strKetQua = "Tháng bảy";  
        break;  
    case 8: strKetQua = "Tháng tám";  
        break;  
    case 9: strKetQua = "Tháng chín";  
        break;  
    case 10: strKetQua = "Tháng mười";  
        break;  
    case 11: strKetQua = "Tháng mười một";  
        break;  
    case 12: strKetQua = "Tháng mười hai";  
        break;  
    default: strKetQua = "Không có tháng phù hợp";  
        break;  
}
```

```
//Hien thi ket qua len giao dien  
txtKetQua.setText(strKetQua);
```

✓ **Một số chú ý về sử dụng Package trong Java**

- **Package** là tập hợp các lớp, giao diện chứa các phương thức, thuộc tính được Java cung cấp sẵn để người lập trình gọi và xử lý công việc của mình.
- **Người lập trình** có thể tạo ra các Package chứa các phương thức hay sử dụng để tái sử dụng cho các dự án sau.

- Để sử dụng tất cả các controls (GUI - Giao diện đồ họa phía người dùng) trong package javax.swing chúng ta có thể khai báo như sau javax.swing.* thay vì khai báo từng lớp cụ thể như:
 - **javax.swing.JButton**: để tạo nút
 - **javax.swing.JLabel**: để tạo nhãn hiển thị text
 - **javax.swing.TextField**: để tạo textbox cho người dùng nhập vào