BÀI GIẢNG

NGÔN NGỮ LẬP TRÌNH JAVA

CHƯƠNG 2:

NHẬP XUẤT VÀ CẦU TRÚC ĐIỀU KHIỂN

TS. Nguyễn Sĩ Thìn (nsthin@vku.udn.vn)



Chương 2. Nhập xuất và cấu trúc điều khiển



Mục tiêu

Hiểu và sử dụng được cách nhập xuất trong Java







Nội dung



Chương 2. Nhập xuất và cấu trúc điều khiển

- 1. Nhập xuất trong Java
- 2. Cấu trúc điều khiển



1. Nhập xuất dữ liệu



Nhập xuất trong Java:

- Đọc dữ liệu vào
- Định dạng dữ liệu ra
- Nhập xuất file

Lưu ý:

- Giao diện "Console"
- Luồng dữ liệu vào "System.in"
- Đối tượng thuộc lớp "Scanner"
- Gói " java.util. * "
- Dùng phương thức "readPassword" của lớp "Console" trong trường hợp đọc mật khẩu

Cú pháp:

```
Scanner in=new Scanner(System.in);
System.out.print("Xin chao! Ten ban La gi? ");
String ten = in.nextLine();
System.out.println("Vui Long nhap mat khau cua ban");
```



1.1 Đọc dữ liệu vào



Một số phương thức cơ bản của lớp Scanner:

STT	java.util.Scanner	Ý nghĩa
1	<pre>String nextLine()</pre>	Đọc một dòng
2	String next()	Đọc một từ
3	<pre>String nextInt()</pre>	Đọc vào một từ thể hiện kiểu số nguyên thuộc kiểu số nguyên Int
4	String nextDouble()	Đọc vào một từ thể hiện kiểu số nguyên thuộc kiểu số thập phân Double
5	Boolean hasNext()	Kiểm tra có hay không từ tiếp theo trong dãy được nhập vào
6	Boolean hasNextInt()	Kiểm tra có hay không từ tiếp theo trong dãy được nhập vào là kiểu số nguyên Int
7	Boolean hasNextDouble()	Kiểm tra có hay không từ tiếp theo trong dãy được nhập vào là kiểu số thập phân Double



1.2 Định dạng dữ liệu ra



Lưu ý:

- Từ SE5 sử dụng phương thức "printf" cho định dạng dữ liệu đầu ra
- Bắt đầu bằng "%" và kết thúc bằng "kí tự" (đại diện cho kiểu dữ liệu của đối số: d, f, s, c, b)
- Sử dụng thêm các ký tự điều khiển như: +, 0, (... và dấu ","

Ví dụ:



1.3 Tập tin đầu vào và tập tin đầu ra



Tập tin đầu vào:

- Dùng đối tượng thuộc lớp Scanner
- Bắt lỗi ngoại lệ
- Ví dụ:

```
Scanner in = new Scanner(Paths.get("myfile.txt"), "UTF-8");
```

- Tập tin đầu ra:
 - Dùng đối tượng thuộc lớp PrintWriter
 - Bắt lỗi ngoại lệ
 - Ví dụ:

```
PrintWriter out = new PrintWriter("myfile.txt", "UTF-8");
```



2. Cấu trúc điều khiển



Khái niệm:

 Khi thực hiện chương trình một số câu lệnh được thực hiện theo một cấu trúc điều khiển chứ không theo trình tự từ trên xuống dưới.

Phân loại:

- Cấu trúc rẽ nhánh: if else; switch
- Cấu trúc vòng lặp: do while, while, for, for each; break, continue



2.1 Lệnh và khối lệnh



Lệnh

- Là một câu lệnh nhằm thực hiện một công việc nào đó.
- · Câu lệnh kết thúc bởi dấu ";"
- ∘ Ví dụ:
 - int i;
 - i++;
- Một cấu trúc điều khiển cũng được xem là 1 lệnh:
 - ✓ Lệnh rẽ nhánh có điều kiện
 - ✓ Lệnh rẽ nhánh không điều kiện
 - ✓ Lệnh lặp

Khối lệnh

- Là dãy các lệnh được đặt giữa cặp ngoặc nhọn "{" và "}"
- Khối lệnh thường được sử dụng khi muốn tập hợp các lệnh thực hiện ứng với một điều kiện hay giá trị nào đó.

```
{
/*Các lệnh*/
}
```

```
if (n1<n2)
{
    min=n1;
    max=n2;
}</pre>
```



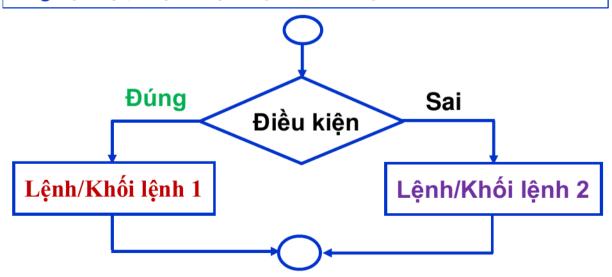
2.2 Lệnh if



• Dạng 1: Lệnh if dạng đầy đủ

<u>Ý nghĩa:</u>

Nếu **Điều kiện** là **đúng** thì thực hiện Lệnh/Khối lệnh 1 Ngược lại, thực hiện Lệnh/Khối lệnh 2.





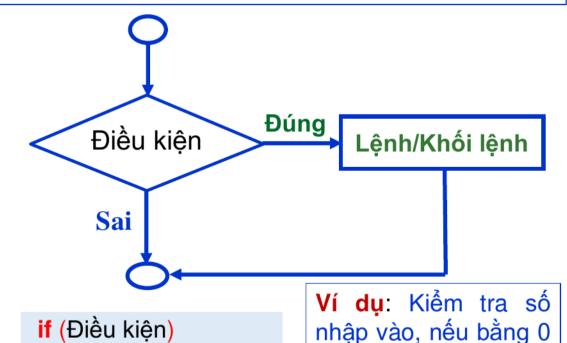
if (Điều kiện)Lệnh/Khối lệnh 1;elseLệnh/Khối lệnh 2;

Ví dụ: Nhập 2 số nguyên và tìm giá trị lớn nhất của hai số.

• Dạng 2: Lệnh if dạng khuyết

Ý nghĩa:

Nếu Điều kiện là **đúng** thì thực hiện **Lệnh/Khối lệnh** Ngược lại, không thực hiện gì cả.



thì xuất thông báo.

10

Lệnh/Khối lệnh;

Lệnh if lồng nhau



Một số lưu ý:

- Có thể sử dụng các toán tử logic để xây dựng các biểu thức điều kiện phức tạp hơn:
 - o Giả sử:

```
if ((đk1 && đk2) || đk3)
```

Ví dụ: Viết biểu thức điều kiện kiểm tra 3 số là 3 cạnh tam giác

```
if ((a>0)&&(b>0)&&(c>0)&&(a<b+c)&&(b<a+c)&&(c<a+b))
```

- Biểu thức điều kiện phải luôn đặt trong trong hai dấu "(" và ")"
- Biểu thức điều kiện có thể là số nguyên hoặc thực
- Sử dụng lệnh if lồng nhau
 - Ví dụ: Chương trình tính nghiệm phương trình ax+b=0

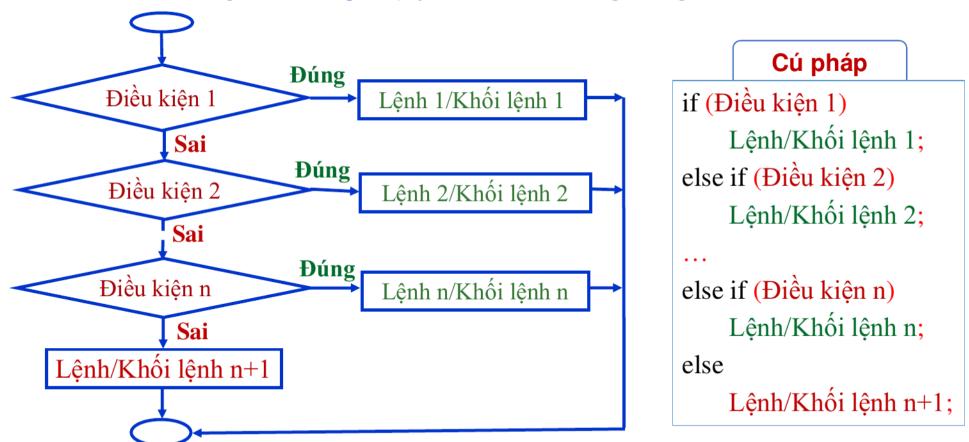


Lệnh if lồng nhau (tt)



Sử dụng else if

• Khi muốn sử dụng một trong n quyết định, sử dụng dạng lệnh if như sau:





• Ví dụ: Chương trình xếp loại kết quả học tập của một sinh viên

2.3 Lệnh switch...case



Lệnh switch ... case cho phép chọn một trong nhiều phương án khác nhau.

```
Cú pháp:
     switch (Biểu thức)
          case n<sub>1</sub>:
                     Các câu lệnh;
          case n<sub>2</sub>:
                     Các câu lệnh;
          case n<sub>k</sub>:
                     Các câu lệnh;
           default:
                                Các câu lệnh;
```

Ý nghĩa:

- Duyệt lần lượt từ trên xuống và kiểm tra xem giá trị của Biểu thức có bằng với giá trị n_i (i từ 1 đến k) đang xét hay không, nếu bằng thì thực hiện các câu lệnh tương ứng.
- Nếu Biểu thức không bằng giá trị nào thì sẽ thực hiện các câu lệnh trong khối default.



2.3 Lệnh switch...case (tt)



Lưu ý:

- Giá trị n_i Phải là 1 trong 2 dạng dữ liệu:
 - Số nguyên
 - o Chuỗi
 - Enumeration
- Thường cuối mỗi dãy lệnh của một nhãn có một lệnh break. Nếu nhảy đến nhãn n_i này, mà kết thúc dãy lệnh trong nhãn này không có câu lệnh break thì máy sẽ tiếp tục thực hiện các lệnh trong nhãn n_{i+1}
- Ví dụ: Sử dụng lệnh switch case nhập vào một sô nguyên trong phạm vi từ 1 đến 3
 và in ra chuỗi giá trị tương ứng.



Bài thực hành chương 2



ĐÈ BÀI

Anh Chị hãy viết chương trình quản lý sinh viên gồm các chức năng sau:

- Cho phép nhập vào danh sách sinh viên
- Hiển thị danh sách sinh viên
- Sắp xếp sinh viên theo điểm trung bình tăng dần

Biết rằng lớp "Student" được kế thừa từ lớp cha "Person":

- + Lớp "Person" gồm các thuộc tính : name, age
- + Lớp "Student" kế thừa lớp Person gồm các thuộc tính bổ sung:

ID (Mã sinh viên), average (điểm trung bình)



THANK YOU!