

## Căn bản về ngôn ngữ lập trình Java

1. Dùng Netbeans (hoặc Eclipse) soạn thảo và thực thi chương trình HelloWorld dùng để hiển thị ra màn hình câu “Hello Java”.
2. Thiết lập các biến môi trường để có thể biên dịch và thực thi chương trình Java bằng cơ chế dòng lệnh. Dùng một trình soạn thảo văn bản bất kỳ để biên soạn chương trình Hello Java ở trên, biên dịch và thực thi chương trình bằng cơ chế dòng lệnh.

Viết chương trình thực hiện các yêu cầu sau:

3. Nhập vào 2 số nguyên, in ra màn hình số nhỏ hơn và số lớn hơn.
4. Nhập vào 3 số nguyên, in ra màn hình số nhỏ nhất và số lớn nhất.
5. Giải và biện luận phương trình bậc 2 ( $ax^2 + bx + c = 0$ ).
6. Nhập vào một số nguyên là năm, cho biết năm đó có là năm nhuận hay không. Biết rằng: năm nhuận là năm chia hết cho 400 hoặc chia hết cho 4 nhưng không chia hết cho 100.
7. Nhập vào một số nguyên là tháng, cho biết tháng đó có bao nhiêu ngày (*nếu là tháng 2 thì yêu cầu nhập thêm năm*).
8. Nhập vào một số thực là điểm ở thang điểm 10, cho biết giá trị ở thang điểm chữ ABCDF. Biết rằng:  $8.5 \rightarrow 10$ : A,  $7.0 \rightarrow 8.4$ : B,  $5.5 \rightarrow 6.9$ : C,  $4.0 \rightarrow 5.4$ : D, dưới 4: F.
9. Nhập vào chỉ số điện cũ và chỉ số điện mới. Tính tiền điện phải trả trong tháng, dữ liệu nhập phải đảm bảo chỉ số điện cũ  $\leq$  chỉ số điện mới. Cách thức tính tiền điện theo quy định sau:
  - Bậc 1: cho các kWh từ 0 đến 50: 1480 VNĐ/kw;
  - Bậc 2: cho các kWh từ 51 đến 100: 1533 VNĐ/kw;
  - Bậc 3: cho các kWh từ 101 đến 200: 1786 VNĐ/kw;
  - Bậc 4: cho các kWh từ 201 đến 300: 2242 VNĐ/kw;
  - Bậc 5: cho các kWh từ 301 trở lên: 2503 VNĐ/kw;
10. Kiểm tra xem ngày (*gồm 3 thành phần là ngày, tháng, năm*) cho trước có phải là một ngày hợp lệ. Trong đó, một ngày được gọi là hợp lệ khi:
  - Năm: số nguyên lớn hơn 0;
  - Tháng: từ 1 đến 12;
  - Ngày từ 1 đến số ngày tối đa của tháng – năm tương ứng.
11. Nhập vào ngày, tháng, năm. Cho biết ngày kế tiếp và ngày kế trước của ngày đó.
12. Nhập số nguyên dương n. Đếm số lượng các ước số của số n.
13. Nhập số nguyên dương n. Liệt kê các ước số lẻ của n.
14. Nhập vào số nguyên dương n. Cho biết n có phải là số nguyên tố hay không?
15. Nhập vào các số thực khác 0 từ bàn phím. Tính tổng các số đã nhập và in ra màn hình. Giả sử quá trình nhập sẽ dừng khi người dùng nhập số 0.
16. Nhập số nguyên dương n. Đếm số lượng các chữ số lẻ của n.
17. Nhập số nguyên dương n. Tính tổng số lượng các chữ số chẵn của n.
18. Nhập 2 số nguyên dương m, n. Xuất ra hình chữ nhật đặc có kích thước mxn dấu \*. Ví dụ: m = 4, n = 5.

```
* * * * *
* * * * *
* * * * *
* * * * *
```

19. Nhập 2 số nguyên dương  $m, n$ . Xuất ra hình chữ nhật rỗng có kích thước  $m \times n$  dấu \*. Ví dụ:  $m = 4, n = 5$ .

```

* * * * *
*       *
*       *
* * * * *

```

20. Nhập số nguyên dương  $h$  là chiều cao. Xuất ra màn hình tam giác cân đặc có chiều cao  $h$ . Ví dụ  $h = 4$ .

```

      *
     * *
    * * *
   * * * *
  * * * * *

```

21. Nhập số nguyên dương  $h$  là chiều cao. Xuất ra màn hình tam giác cân rỗng có chiều cao  $h$ . Ví dụ  $h = 4$ .

```

      *
     *   *
    *     *
   *       *
  * * * * *

```

22. Viết chương trình tạo mảng một chiều và cho phép người dùng chọn thực hiện các chức năng bên dưới, đến khi nào người dùng chọn 0 thì thoát chương trình:

- 1. Nhập giá trị cho  $n$  phần tử mảng từ bàn phím;
- 2. Phát sinh giá trị ngẫu nhiên từ -199 đến 199 cho  $n$  phần tử mảng;
- 3. Xuất mảng ra màn hình;
- 4. Liệt kê các giá trị âm có trong mảng;
- 5. Liệt kê các số nguyên tố có trong mảng;
- 6. Liệt kê các phần tử có giá trị nằm trong đoạn  $[a, b]$  cho trước;
- 7. Tính tổng giá trị các phần tử là số nguyên tố;
- 8. Tính trung bình cộng của các phần tử dương có trong mảng;
- 9. Đếm số phần tử có giá trị lớn hơn  $x$  cho trước;
- 10. Đếm số phần tử có giá trị là số nguyên tố trong mảng;
- 11. Kiểm tra mảng có phải là mảng chứa toàn số nguyên tố;
- 12. Kiểm tra mảng có phải là mảng tăng dần;
- 13. Tìm giá trị lớn nhất trong mảng;
- 14. Tìm giá trị nhỏ nhất trong mảng;
- 15. Tìm số âm lớn nhất trong mảng;
- 16. Đảo ngược mảng.

23. Viết chương trình tạo một mảng hai chiều (ma trận) và thực hiện các yêu cầu sau:

- 1. Nhập giá trị cho  $m \times n$  phần tử của mảng từ bàn phím;
- 2. Phát sinh giá trị ngẫu nhiên từ -99 đến 99 cho  $m \times n$  phần tử của mảng;
- 3. Xuất giá trị các phần tử ra màn hình;
- 4. Tính tổng giá trị các số lẻ;
- 5. Tính tổng các giá trị dương trên dòng  $k$  cho trước;
- 6. Đếm số lượng số dương;
- 7. Đếm số lượng số nguyên tố trên một dòng của ma trận;
- 8. Kiểm tra ma trận có tồn tại số âm hay không;

- 9. Kiểm tra các phần tử trên dòng k cho trước có tăng dần từ trái qua phải hay không;
- 10. Liệt kê dòng có chứa số nguyên tố;
- 11. Liệt kê dòng chứa toàn giá trị chẵn.

**24. Cài đặt lớp **Diem2D** (Điểm trong không gian 2 chiều) gồm:**

- Thuộc tính: x, y là số nguyên.
- Các phương thức bao gồm :
  - + Phương thức khởi tạo mặc định: `Diem();`
  - + Phương thức khởi tạo có 2 tham số: `Diem(int, int );`
  - + Nhập tọa độ cho điểm từ bàn phím `void nhapDiem ();`
  - + Hiển thị ra màn hình tọa độ điểm theo dạng (x,y): `void hienThi();`
  - + Dời điểm đi 1 độ dời (dx, dy): `void doiDiem(int dx, int dy);`
  - + Lấy ra giá trị hoành độ của điểm: `int giaTriX();`
  - + Lấy ra giá trị tung độ của điểm: `int giaiTriY();`
  - + Tính khoảng cách từ điểm đó đến gốc tọa độ: `float khoangCach();`
  - + Tính khoảng cách từ điểm đó đến 1 điểm khác: `float khoangCach(Diem d);`
- **Viết lớp **SDDiem** có chứa hàm **main()** khai thác lớp vừa định nghĩa:**
  - + Tạo ra điểm A tọa độ (3,4). Hiển thị tọa độ điểm A ra màn hình.
  - + Tạo ra điểm B với giá trị nhập từ bàn phím. Hiển thị tọa độ điểm B ra màn hình.
  - + Tạo ra điểm C đối xứng với điểm B qua gốc tọa độ. Hiển thị tọa độ điểm C ra màn hình.
  - + Hiển thị ra màn hình khoảng cách từ điểm B đến tâm O.
  - + Hiển thị ra màn hình khoảng cách từ điểm A đến điểm B.

**25. Cài đặt lớp **PhanSo** (Phân số) gồm:**

- Các thuộc tính: tử số và mẫu số có kiểu số nguyên.
- Các hàm xây dựng gồm:
  - + Hàm khởi tạo mặc định:
  - + Hàm khởi tạo gồm 2 tham số:
- Các phương thức khác bao gồm:
  - + Hàm nhập giá trị cho 1 phân số. Nếu phân số vừa nhập có mẫu số = 0 thì yêu cầu nhập lại.
  - + Hàm hiển thị phân số theo dạng tu/mau hoặc -tu/mau. Yêu cầu: nếu tử số =0 thì chỉ in ra số 0, nếu mẫu số =1 thì chỉ in ra tử số.
  - + Hàm nghịch đảo phân số (làm thay đổi giá trị phân số) `void nghichDao();`
  - + Hàm tính ra phân số nghịch đảo của 1 phân số (phân số sẽ giữ nguyên nhưng hàm trả về giá trị là phân số nghịch đảo của nó). `PhanSo giaTriNghichDao();`
  - + Hàm tính ra giá trị thực của phân số. Chẳng hạn phân số 1/2 có giá trị là 0.5.
  - + Hàm so sánh lớn hơn với phân số a. `boolean lonHon(PhanSo a);`
  - + Hàm cộng, trừ, nhân, chia phân số với 1 phân số a. Kết quả của hàm là 1 phân số.

- + Hàm cộng, trừ, nhân, chia phân số với 1 số nguyên. Kết quả của hàm là 1 phân số.
- Viết lớp SDPhanSo có chứa hàm main() sử dụng lớp PhanSo :
  - + Tạo phân số  $a = 3/7$  ,  $b = 4/9$  . Hiện thị giá trị của chúng ra màn hình.
  - + Tạo 2 phân số x và y. Nhập giá trị cho x và y từ bàn phím.
  - + Hiện thị giá trị nghịch đảo của phân số x ra màn hình (*không làm thay đổi giá trị của x*).
  - + Tính tổng của  $x + y$  và in kết quả ra màn hình.
  - + Nhập vào 1 danh sách gồm n phân số (n : nhập từ bàn phím ).
  - + Tính tổng n phân số đó.
  - + Tìm phân số lớn nhất trong danh sách phân số trên.
  - + Sắp xếp danh sách phân số theo thứ tự tăng dần.
- 26. Xây dựng lớp TamGiac (*tam giác trong không gian oxy*) có các thuộc tính và phương thức như sau:
  - **Thuộc tính:** 3 đỉnh là 3 điểm trong không gian oxy.
  - **Phương thức:**
    - + Phương thức khởi tạo mặc định
    - + Phương thức khởi tạo có đầy đủ tham số
    - + Phương thức khởi tạo sao chép
    - + Các phương thức get/set
    - + Phương thức nhập
    - + Phương thức xuất
    - + Phương thức tính diện tích
    - + Phương thức tính chu vi
  - Xây dựng lớp Main có phương thức main, suy nghĩ kịch bản sử dụng các phương thức đã cài đặt của lớp TamGiac.
- 27. Xây dựng lớp DaGiac (*đa giác trong không gian oxy*) có các thuộc tính và phương thức sau:
  - **Thuộc tính:** danh sách các đỉnh là các điểm trong không gian oxy
  - **Phương thức:**
    - + Phương thức khởi tạo mặc định
    - + Phương thức khởi tạo có đầy đủ tham số
    - + Phương thức khởi tạo sao chép
    - + Các phương thức get/set
    - + Phương thức nhập
    - + Phương thức xuất
    - + Phương thức tính chu vi
    - + Phương thức tìm đỉnh xa gốc tọa độ nhất

Lưu ý: sử dụng 2 cách (*mảng thông thường và ArrayList*).

*Xây dựng phương thức main, suy nghĩ kịch bản sử dụng các phương thức đã cài đặt của lớp DaGiac.*

**28.** Cài đặt lớp **DiemMau** (Điểm có màu) thừa kế từ lớp Diem bổ sung thêm:

- Thuộc tính: màu (kiểu String)
- Các phương thức:
  - + Hàm xây dựng: DiemMau(int x, int y, String mau)
  - + Hàm gán giá trị màu cho điểm: void GanMau(String mau)
  - + Hàm nhập, hàm hiển thị thông tin
- Viết lớp SDDiemMau có hàm main() thực hiện các công việc sau:
  - + Tạo 1 điểm màu A có tọa độ là (5, 10) và màu là trắng. Hiển thị thông tin ra màn hình.
  - + Tạo 1 điểm màu tổng quát B. Nhập giá trị từ bàn phím cho điểm B. Dời điểm B đi 1 độ dời (10,8). Hiển thị tọa độ điểm B ra màn hình. Gán màu mới cho điểm B là màu “Vàng”.

**29.** Công ty ABC là công ty sản xuất kinh doanh thú nhồi bông. Công ty có nhiều nhân viên làm việc trong 3 bộ phận khác nhau gồm: bộ phận quản lý, bộ phận sản xuất, bộ phận văn phòng. Việc tính lương cho nhân viên dựa vào các thông tin sau:

- Đối với nhân viên văn phòng:

$$\text{Lương} = \text{LCB} + \text{Số ngày LV} * 100.000 + \text{Trợ cấp}$$

- Đối với nhân viên sản xuất:

$$\text{Lương} = \text{LCB} + \text{Số sản phẩm} * 2.000$$

- Đối với nhân viên quản lý:

$$\text{Lương} = \text{LCB} * \text{Hệ số chức vụ} + \text{Thưởng}$$

Hiện tại công ty áp dụng mức LCB là 1,490,000.

Ngoài ra, công ty cần quản lý các thông tin về nhân viên của mình như: họ tên, năm vào làm và các thông số trên để tính lương của nhân viên trong công ty. Hãy thiết kế các lớp thích hợp để thực hiện các yêu cầu sau:

1. Nhập thông tin của các nhân viên để phục vụ cho việc tính lương.
2. Xuất thông tin của các nhân viên.
3. Tính tổng lương của công ty và tổng lương của từng loại nhân viên.

**30.** Một trường học cần xây dựng ứng dụng quản lý điểm các môn học. Biết rằng mỗi môn học đều có các thông tin sau: Mã môn học, tên môn học, số tín chỉ. Các môn học được chia làm 3 loại khác nhau:

- **Môn lý thuyết:** có 3 cột điểm là điểm tiểu luận và điểm cuối kỳ.  $\text{ĐTB} = \text{TL} * 0.2 + \text{GK} * 0.3 + \text{CK} * 0.5$ .
- **Môn thực hành:** có 4 cột điểm kiểm tra. ĐTB sẽ được tính bằng trung bình cộng các bài kiểm tra.
- **Môn đồ án:** có điểm của giáo viên hướng dẫn (GVHD) và giáo viên phản biện đồ án (GVPB). ĐTB sẽ được tính là trung bình cộng điểm của GVHD và GVPB.

Hãy xây dựng ứng dụng quản lý điểm theo mô tả như trên cho phép nhập xuất danh sách môn học và tính điểm trung bình các môn học.

**31.** Công ty ABC cần quản lý 4 loại đối tượng bất động sản là: đất trống, nhà ở, biệt thự và khách sạn. Biết rằng tất cả các đối tượng này đều có các thông tin: mã số, chiều dài, chiều rộng và phương thức tính giá trị. Biết rằng giá trị được tính như sau:

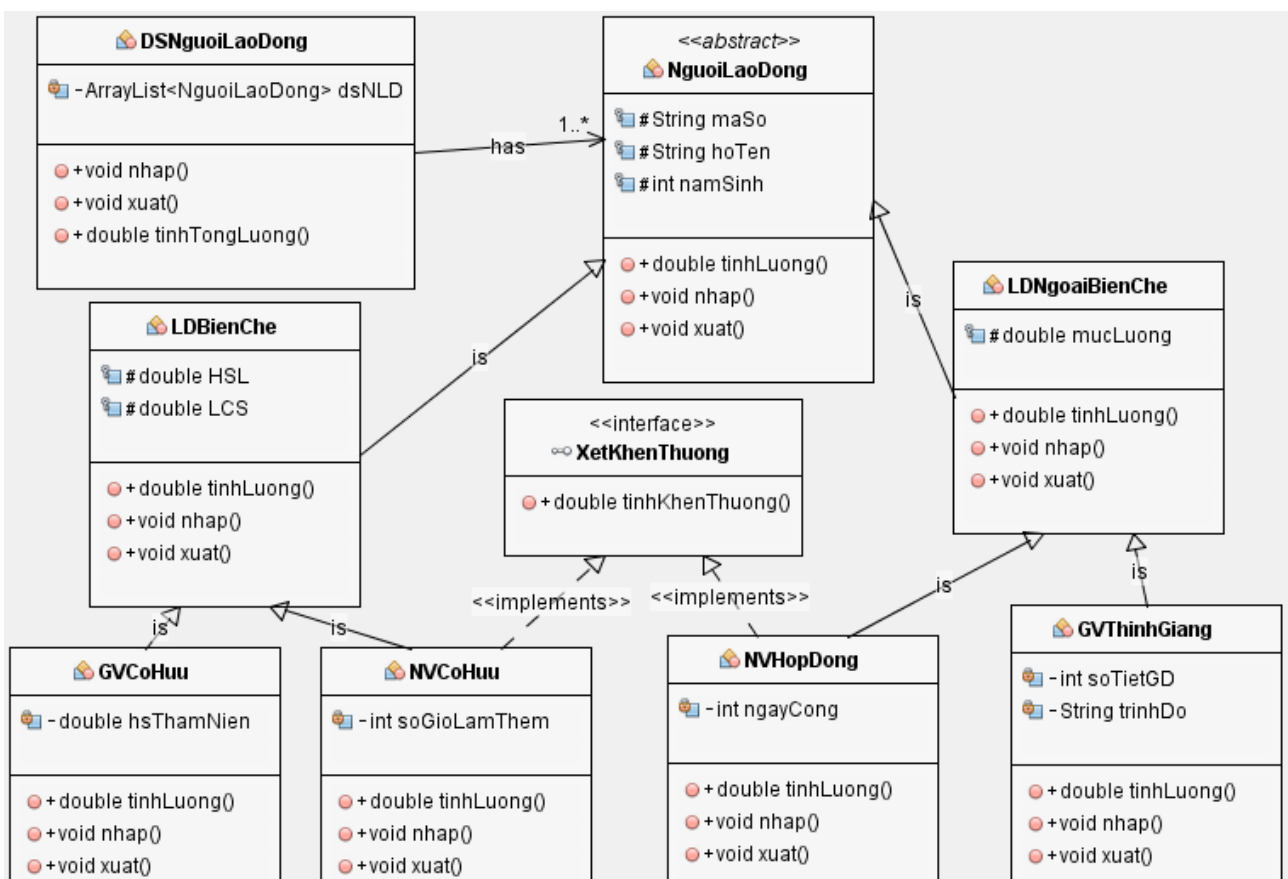
- Đất trống: giá bán = diện tích \* 10.000.
- Nhà ở có thêm thông tin số lầu: giá bán = diện tích \* 10.000 + số lầu \* 100.000.
- Khách sạn có thêm thông tin về số sao: giá bán = 100.000 + số sao \* 50.000 .
- Biệt thự: giá bán = diện tích \* 400.000.

Tuy nhiên trong 4 loại bất động sản trên thì có 2 loại sản phẩm khi mua khách hàng phải đóng thêm phí kinh doanh là biệt thự và khách sạn. Biết rằng, phí kinh doanh sẽ được tính như sau:

- Biệt thự: Diện tích \* 1000
- Khách sạn: Chiều rộng \* 5000

Hãy xây dựng ứng dụng quản lý bất động sản theo mô tả như trên cho phép nhập xuất danh sách và tính giá bán của từng loại bất động sản và tổng tất cả bất động sản.

**32.** Xây dựng chương trình theo sơ đồ lớp sau:



### Cách tính lương:

- Lương GV cơ hữu =  $HSL * LCS * (1 + hsThamNien)$
- Lương NV cơ hữu =  $HSL * LCS + soGioLamThem * 50000$
- Lương NV hợp đồng =  $mucLuong * ngayCong$

- Lương GV thỉnh giảng =  $mucLuong * soTietGD * trinhDo$ . Trong đó: ĐH: 1.0; ThS: 1.2; TS: 1.5; PGS: 1.8; GS: 2.0.

*Trong đó:* LCS giống nhau cho tất cả GV cơ hữu và NV cơ hữu, hiện tại là 1490000.

Xét khen thưởng đối NV cơ hữu và NV hợp đồng:

- NV cơ hữu: nếu soGioLamThem từ 40 trở lên thì thưởng thêm 1000000.
- NV hợp đồng: nếu ngayCong từ 25 trở lên thì thưởng thêm 1500000.

**33.** Xây dựng ứng dụng quản lý khách hàng dùng Text File theo yêu cầu sau:

- Thông tin khách hàng: mã số, họ tên, năm sinh
- Nhập danh sách khách hàng từ bàn phím
- Xuất danh sách khách hàng ra màn hình
- Ghi danh sách khách hàng ra file
- Đọc danh sách khách hàng từ file

**34.** Xây dựng ứng dụng quản lý danh bạ dùng Serialize File theo yêu cầu sau:

- Thông tin danh bạ: tên, số phone
- Nhập danh sách danh bạ từ bàn phím
- Xuất danh sách danh bạ ra màn hình
- Ghi danh sách danh bạ vào file
- Đọc danh sách danh bạ từ file

**35.** Chỉnh sửa các bài tập từ 1.29 đến 1.32 để thực hiện nhập dữ liệu từ file và xuất kết quả ra file.

---Hết phần 1---