

## Chương 1. Căn bản về ngôn ngữ lập trình Java

- 1.1. Dùng Netbeans (hoặc Eclipse) soạn thảo và thực thi chương trình HelloWorld dùng để hiển thị ra màn hình câu “Hello Java”.
- 1.2. 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 HelloWorld ở trên, biên dịch và thực thi chương trình bằng cơ chế dòng lệnh.
- 1.3. Nhập vào 2 số nguyên, in ra màn hình số nhỏ hơn và số lớn hơn.
- 1.4. Nhập vào 3 số nguyên, in ra màn hình số nhỏ nhất và số lớn nhất.
- 1.5. Giải và biện luận phương trình bậc 2 ( $ax^2 + bx + c = 0$ ).
- 1.6. 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.
- 1.7. 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*).
- 1.8. 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.
- 1.9. 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;
- 1.10. 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.
- 1.11. Nhập vào ngày, tháng, năm. Cho biết ngày kế tiếp và ngày kế trước của ngày đó.
- 1.12. Nhập số nguyên dương n. Đếm số lượng các ước số của số n.
- 1.13. Nhập số nguyên dương n. Liệt kê các ước số lẻ của n.
- 1.14. Nhập vào số nguyên dương n. Cho biết n có phải là số nguyên tố hay không?
- 1.15. 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.
- 1.16. Nhập số nguyên dương n. Đếm số lượng các chữ số lẻ của n.
- 1.17. Nhập số nguyên dương n. Tính tổng số lượng các chữ số chẵn của n.
- 1.18. Nhập 2 số nguyên dương m, n. Xuất ra hình chữ nhật đặc có kích thước m x n dấu \*. Ví dụ: m = 4, n = 5.

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

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

```

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

```

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

```

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

```

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

```

      *
     *   *
    *     *
   *       *
  *         *

```

**1.22.** Viết chương trình tạo mảng một chiều tối đa 500 phần tử (*giá trị mỗi phần tử là số nguyên từ  $-10^6$  đến  $10^6$* ) 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 ( $n \leq 500$ );
- 2. Phát sinh giá trị ngẫu nhiên từ -199 đến 199 cho  $n$  phần tử mảng ( $n \leq 500$ );
- 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.

**1.23.** Viết chương trình tạo một mảng hai chiều (ma trận) tối đa  $50 \times 100$  phần tử (*giá trị mỗi phần tử là số nguyên từ  $-10^6$  đến  $10^6$* ) và thực hiện các yêu cầu sau:

- 1. Nhập giá trị cho  $m \times n$  phần tử của mảng từ bàn phím ( $m \leq 50, n \leq 100$ );
- 2. Phát sinh giá trị ngẫu nhiên từ -99 đến 99 cho  $m \times n$  phần tử của mảng ( $n \leq 100$ );
- 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;