
LẬP TRÌNH CƠ BẢN VỚI JAVA

Lưu ý về phần bài tập lập trình

- *Tất cả các dữ liệu đầu vào đều được nhập theo cách lấy từ tệp văn bản*
 - *Tất cả các chức năng được yêu cầu trong mỗi chương trình phải được viết thành chương trình con riêng (là hàm), hàm main chỉ làm việc nhận giá trị từ đối dòng lệnh và gọi tới các chương trình con.*
1. Viết chương trình nhập vào một mảng số nguyên a gồm n phần tử. Biến đổi mảng sao cho các phần tử trong mảng được đổi chỗ cho nhau theo thứ tự ngược lại: phần tử đầu tiên thành phần tử cuối cùng và ngược lại v.v.
 2. * Viết chương trình nhập vào một mảng số nguyên a gồm n phần tử. Tìm dãy số nguyên liên tiếp dài nhất xuất hiện trong a thoả mãn các phần tử trong dãy có giá trị bằng nhau. In ra màn hình chiều dài của dãy dài nhất và giá trị của phần tử trong dãy.
Ví dụ: nếu mảng $a = \{1\ 2\ 2\ 1\ 5\ 1\ 1\ 7\ 7\ 7\ 7\ 1\ 1\}$ thì chương trình sẽ in ra màn hình hai số tương ứng là 4 7
 3. Viết chương trình `WordCount.java` nhập vào một chuỗi kí tự, đếm và in ra số các từ xuất hiện trong chuỗi.

-
4. **Hình vuông ma thuật***. Viết chương trình `MagicSquare.java` đọc vào một số nguyên lẻ n và in ra màn hình một *hình vuông ma thuật* có kích thước $n \times n$. Hình vuông ma thuật là hình vuông chứa các số nguyên từ 1 đến n^2 , mỗi số một lần thoả mãn tổng các phần tử trên các hàng, cột và đường chéo là bằng nhau.

4	9	2		11	18	25	2	9
3	5	7		10	12	19	21	3
8	1	6		4	6	13	20	22
				23	5	7	14	16
				17	24	1	8	15

Gợi ý: Sắp xếp các số nguyên từ 1 đến n^2 theo thứ tự tăng dần. Số đầu tiên được gán vào ô có vị trí tại hàng cuối cùng, cột ở giữa. Lặp quá trình gán số nguyên tiếp theo vào ô hàng xóm liền kề theo đường chéo, phía dưới, bên phải. Nếu ô này đã được gán số thì sử dụng ô liền kề phía trên. Sử dụng kỹ thuật "quấn quanh" để xử lý trường hợp biên.