

BÀI THỰC HÀNH 2

Nội dung:

- Các kỹ thuật lập trình cơ bản (lính canh, cờ hiệu)
- Thao tác trên tập tin

Bài 2.1 Viết các hàm thực hiện các yêu cầu sau:

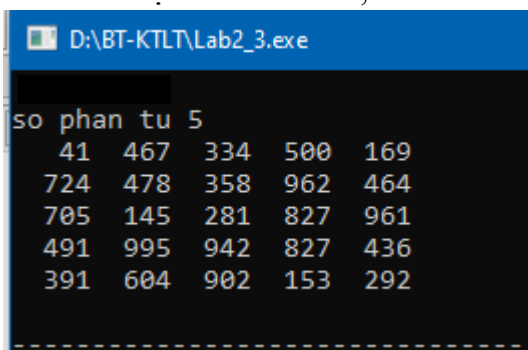
- Nhập vào dãy số nguyên gồm n phần tử ($3 < n < 50$)
- Tìm số âm đầu tiên trong dãy số, nếu có in ra vị trí và giá trị (dùng kỹ thuật lính canh)
- Kiểm tra có phần tử có giá trị x trong mảng hay không (dùng kỹ thuật lính canh)
- Sử dụng kỹ thuật cờ hiệu để kiểm tra dãy số có giảm dần hay không?
- Sử dụng kỹ thuật cờ hiệu để kiểm tra dãy số có toàn là số nguyên dương hay không?
- Hàm `main()` gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 2.2 Viết các hàm thực hiện các yêu cầu sau:

- Tạo n phần tử là số nguyên ngẫu nhiên trong $[0, 1000]$ với $2 \leq n \leq 30$, ghi n số nguyên vừa tạo vào file “DaySo.inp”;
- Đọc n số nguyên từ file vào chương trình;
- Xuất dãy số vừa đọc được ra màn hình;
- Tính và in ra tổng các số nguyên dương chẵn trong dãy số;
- Hàm `main()` gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 2.3 Viết các hàm thực hiện các yêu cầu sau:

- Nhập ma trận gồm $n \times n$ số nguyên với $2 \leq n \leq 20$, và ghi ma trận vào file “MaTran.inp”;
- Đọc dữ liệu từ file “MaTran.inp” ra lưu vào ma trận A ;
- Xuất ma trận ra màn hình;



- Tính tổng các số nằm trên đường chéo chính và ghi vào file “KetQua.out”;
- Viết macro tìm số lớn nhất trong 2 số;
- Tìm số lớn nhất (sử dụng macro) trong ma trận và ghi vào file “KetQua.out”.
- Hàm `main()` gọi các hàm trên thực hiện để kiểm tra kết quả.

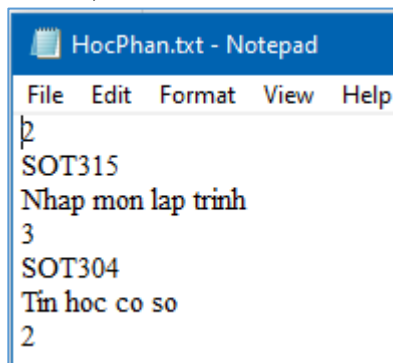
Bài 2.4 Viết các hàm thực hiện các yêu cầu sau:

- Tạo file “DaySo2_4.txt” có cấu trúc:
 - Dòng đầu tiên ghi số lượng phần tử của dãy số $3 \leq n \leq 40$;

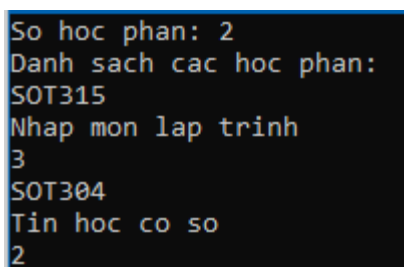
- Dòng tiếp theo ghi giá trị của các phần tử trong dãy số, các phần tử là số thực, mỗi phần tử cách nhau bởi ít nhất một khoảng trắng.
- b) Đọc dữ liệu từ file “DaySo2_4.txt” vào chương trình và lưu vào mảng a;
- c) Xuất dãy số ra màn hình;
- d) Đếm và trả về số lượng phần tử có giá trị nhỏ hơn x, với x là tham số của hàm;
- e) Hàm main() gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 2.5 Viết chương trình thực hiện các yêu cầu sau:

- a) Khai báo kiểu dữ liệu cấu trúc để biểu diễn thông tin của học phần. Biết rằng một học phần gồm có: mã học phần, tên học phần, số tín chỉ.
- b) Tạo file văn bản bằng NotePad và nhập dữ liệu cho n học phần lưu với tên HocPhan.txt ($2 \leq n < 50$)



- c) Đọc n học phần từ file HocPhan.txt vào chương trình và lưu vào mảng học phần.
- d) Xuất ra màn hình danh sách học phần vừa đọc ra từ file.



Bài 2.6* Viết các hàm thực hiện các yêu cầu sau:

- a) Nhập vào dãy số nguyên gồm n phần tử ($3 < n < 50$)
- b) Tìm số chẵn đầu tiên trong dãy số (dùng kỹ thuật lỉnh canh)
- c) Sử dụng kỹ thuật cờ hiệu để kiểm tra dãy số có giảm dần hay không?
- d) Hàm main() gọi các hàm trên thực hiện để kiểm tra kết quả.

Bài 2.7* Viết chương trình thực hiện các yêu cầu sau:

- a) Tạo file văn bản bằng NotePad và nhập dữ liệu là n số nguyên, lưu file với tên DaySoNguyen.txt ($2 \leq n < 50$);
- b) Đọc n số nguyên từ file vào chương trình và lưu trong biến mảng a;
- c) Xuất mảng a ra màn hình;
- d) Tính và trả về tổng các phần tử là số nguyên tố trong dãy số;

Bài 2.8* Viết chương trình thực hiện các yêu cầu:

- a) Tạo cấu trúc Sinh viên gồm các thông tin: mã số SV, họ tên sinh viên, năm sinh;
- b) Viết chương trình nhập vào n Sinh viên ($2 \leq n < 50$), và ghi thông tin n Sinh viên vào file nhị phân *SinhVien.inp*;
- c) Đọc thông tin Sinh viên thứ p từ file và hiển thị ra màn hình, với p được nhập vào từ bàn phím;

----- hết -----