

## Đề thi Thực hành Cuối kỳ

Môn: Kỹ thuật lập trình

**Bài 1** – Viết chương trình quản lý sinh viên. Mỗi sinh viên có các thông tin sau:

- Mã số sinh viên (chuỗi gồm 8 ký tự)
- Họ và tên (chuỗi gồm tối đa 50 ký tự)
- Ngày sinh (kiểu Ngày)
- Giới tính (kiểu boolean)
- Điểm trung bình (số thực, không âm)

Cấu trúc dữ liệu để quản lý thông tin sinh viên dưới dạng danh sách liên kết đơn được định nghĩa như sau:

```
struct Ngay {
    int ng, th, nam;
};

struct SINHVIEN {
    char maso[9];           // "19120100",
    char hoten[50];         // "Nguyen Van Tung"
    bool GioiTinh;          // 1: "Nam"; 0: "Nu"
    Ngay NgaySinh;          // 01/01/2000
    float Diem;             // 6.75
};

struct Node {
    SINHVIEN data;
    Node* Next;
};

struct List {
    Node* Head;
    Node* Tail;
};
```

**Yêu cầu:**

- Hãy viết hàm cho phép nhập thông tin của một sinh viên và đưa thêm vào danh sách liên kết.
- Hãy viết hàm xuất ra màn hình thông tin của tất cả những sinh viên trong danh sách.
- Giả sử danh sách sinh viên đã có thứ tự tăng dần theo mã số sinh viên, viết hàm thêm một sinh viên vào danh sách, mà vẫn duy trì thứ tự sau khi thêm.
- Viết hàm tìm tất cả các sinh viên có cùng ngày sinh nhật trong ngày hiện tại.
- Viết hàm tìm tất cả các sinh viên có điểm trung bình dưới 5
- Viết hàm xuất toàn bộ danh sách sinh viên có điểm trung bình dưới 5 ra tập tin văn bản
- Viết hàm đọc danh sách các sinh viên có điểm trung bình dưới 5 từ tập tin và xuất ra màn hình.

h) Viết hàm xóa tất cả các sinh viên có điểm trung bình dưới 5 khỏi danh sách sinh viên.

**Bài 2** Cho tổng  $F(x, n) = x + x^2/2! + x^3/3! + \dots + x^n/n!$ . Với  $x$  là số thực,  $n$  nguyên không âm.

1. Viết công thức tính  $F(x, n)$  theo  $F(x, n-1)$  và  $F(x, n-2)$ , Viết hàm tính  $F(x, n)$  sử dụng đệ qui.
2. Viết hàm tính lũy thừa:  $x^n$ .
3. Viết hàm tính giai thừa của  $n$ :  $n!$
4. Viết hàm tính tổng  $F(x, n)$  không sử dụng đệ qui.

Lưu ý: Câu 2,3 sử dụng cho câu 4.

**Bài 3** Cho hàng đợi ưu tiên được cài đặt dùng danh sách liên kết, mỗi node của danh sách liên kết trong cài đặt hàng đợi ưu tiên có ba thành phần:

- key: lưu giá trị số nguyên.
- Priority: lưu độ ưu tiên.
- Address: lưu địa chỉ của node kế tiếp

Chú ý: Phần tử có độ ưu tiên cao nhất sẽ được thực thi trước phần tử có độ ưu tiên thấp hơn, trong Hình 2.1 phần tử có độ ưu tiên cao nhất là 0 và phần tử có độ ưu tiên thấp nhất là 9, nếu hai phần tử có cùng độ ưu tiên thì chúng sẽ được thực thi theo trình tự chúng được thêm vào danh sách.

Danh sách liên kết cài đặt hàng đợi được khai báo như sau:

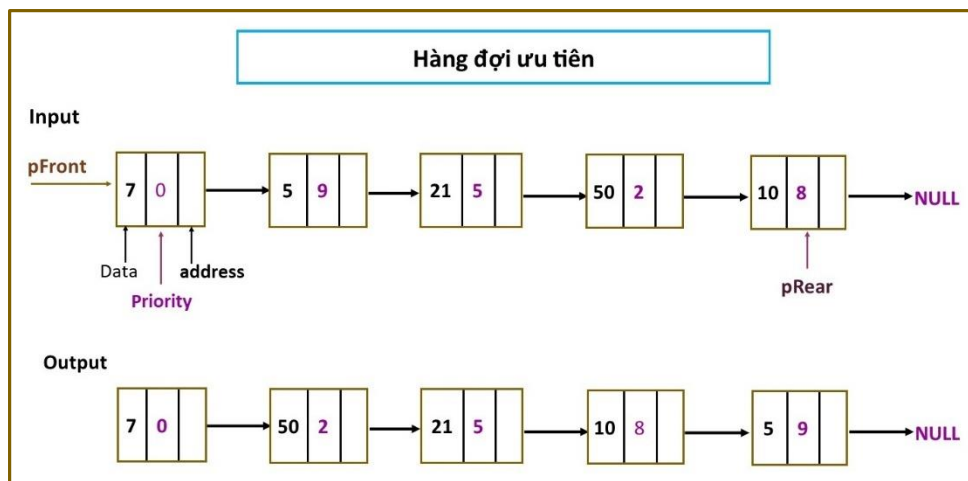
```
/*----- Khai bao Queue -----*/
struct Element {
    int key;
    int priority; };

/* Node of Queue*/
struct NODE {
    Element data;
    NODE *pNext;};

/* Queue */
struct QUEUE {
    NODE *pFront;
    NODE *pRear;};
```

Yêu cầu

- a) Viết hàm tạo hàng đợi ưu tiên như minh họa ở Hình 2.1.
- b) Viết hàm xuất hàng đợi ưu tiên ra màn hình.
- c) Viết hàm xuất hàng đợi ưu tiên ra tập tin.
- d) Viết hàm đọc hàng đợi ưu tiên từ tập tin.
- e) Viết hàm lấy một phần tử ra khỏi hàng đợi.



Hình 2.1: Minh họa tạo hàng đợi ưu tiên

**Lưu ý:** Các sinh viên có bài giống nhau sẽ bị không điểm.

\*\*\*