

# Bài tập danh sách móc nối

## BÀI TẬP 3

### DANH SÁCH LIÊN KẾT

Cho danh sách liên kết đơn có nút đầu trỏ bởi biến con trỏ F (gọi tắt là danh sách F) có khai báo như sau:

```
struct Node
{
    int Info; //trường Info lưu thông tin của mỗi nút
    //là một số nguyên
    Node *Next; //trường Next lưu địa chỉ nút tiếp
    //theo.
};
Node *F;
```



Viết các chương trình con sau bằng phương pháp lặp và phương pháp đệ quy:

1. Tìm số nút trong danh sách F có giá trị trường Info bé hơn một số nguyên X cho trước.
2. Chèn một nút có giá trị trường Info bằng X vào danh sách F đã được sắp xếp theo thứ tự tăng dần của trường Info.
3. Xóa một nút có giá trị trường Info bằng X của danh sách F đã được sắp xếp theo thứ tự tăng của trường Info.
4. Tìm giá trị lớn nhất của trường Info của các nút thuộc danh sách F, với giả thiết rằng danh sách F có ít nhất là một nút (F khác NULL).

5. Bổ sung một nút mới có giá trị trường Info bằng X vào cuối danh sách F.
6. Xoá nút cuối của danh sách F.
7. Tìm địa chỉ của một nút thuộc danh sách F có giá trị trường Info bằng X (nếu có), hoặc trả về giá trị NULL nếu tìm không có, trong 2 trường hợp sau: a) Danh sách F không được sắp xếp. b) Danh sách F được sắp xếp theo thứ tự tăng dần của trường Info.
8. In giá trị trường Info các nút của danh sách F theo thứ tự ngược (từ nút cuối đến nút đầu).
9. Tạo ra một danh sách mới L (nút đầu trỏ bởi L) có dữ liệu trường Info lần lượt được sao chép từ danh sách F.
10. Cho trước 2 danh sách F1 và F2 đã được sắp xếp theo thứ tự tăng của trường Info và lần lượt biểu diễn cho 2 tập hợp các số nguyên F1 và F2. Viết hàm nhằm tạo ra danh sách F3 để biểu diễn cho: a) Hợp của 2 tập hợp F1 và F2. Ví dụ: Nếu  $F1 = \{1,3,5\}$  và  $F2 = \{1,2,5\}$  thì  $F3 = \{1,2,3,5\}$ . b) Giao của 2 tập hợp F1 và F2. Ví dụ: Nếu  $F1 = \{1,3,5\}$  và  $F2 = \{1,2,5\}$  thì  $F3 = \{1,5\}$ .