

I. Bài tập danh sách liên kết đơn

1. Trình bày từng bước các thao tác trên danh sách liên kết đơn
2. Viết chương trình thực hiện việc sắp xếp danh sách liên kết đơn bao gồm các phần tử là số nguyên.
3. Viết chương trình cộng 2 đa thức được biểu diễn thông qua danh sách liên kết đơn.
4. Hãy cài đặt chương trình cho phép nhập vào một biểu thức bao gồm: các số, các toán tử +, -, *, /, div (chia dư) và các hàm toán học sin, cos, tan, ln, e^x , trong biểu thức có các dấu mở, đóng ngoặc "(", ")" và chương trình sẽ tính toán giá trị của biểu thức này.
5. Cho danh sách liên kết được mô tả bởi cấu trúc dữ liệu trên C như sau:

```
struct Node  
{ int info;  
  struct Node *next;  
};
```

Hãy viết hàm void Shuffle (Node *list) nhận đầu vào là danh sách liên kết với phần tử đầu tiên được trỏ bởi list, thực hiện các công việc sau đây:
 - a. Sắp xếp lại các phần tử của danh sách đã cho, sao cho các nút chẵn đứng trước các nút lẻ và trong trường hợp ngược lại, thứ tự tương đối ban đầu của các nút là không thay đổi. Một nút được gọi là nút chẵn hay lẻ nếu nó đứng ở vị trí chẵn hay lẻ trong danh sách (vị trí của các nút trong danh sách được đánh số từ phần tử đầu tiên đến phần tử cuối cùng bắt đầu từ 0).
 - b. Đưa ra màn hình danh sách thu được.
Ví dụ, nếu danh sách đã cho là (11, 13, 7, 9, 3, 10) thì kết quả hiển thị lên màn hình là (13, 9, 10, 11, 7, 3).
6. Giả sử cho một danh sách liên kết đơn có thành phần dữ liệu là các số nguyên dương, người ta muốn tách danh sách đã cho thành hai danh sách riêng biệt, trong đó một danh sách lưu số chẵn, một danh

sách lưu số lẻ. Hãy trình bày giải thuật và thực hiện cài đặt để tách danh sách đã cho sao cho hiệu quả nhất về thời gian xử lý và bộ nhớ sử dụng, đặc biệt xét cả trong trường hợp danh sách đã cho bao gồm tất cả là số chẵn hoặc số lẻ.

7. Hãy trình bày giải thuật và thực hiện cài đặt trộn hai danh sách liên kết đơn đã có thứ tự (tăng hoặc giảm dần) thành một danh sách có thứ tự sao cho tối ưu bộ nhớ nhất có thể.