**Danh sách liên kết đơn**

-----------------------------------------------------------------------------

Element:

Element

Null

Data

pointer

struct Element{

int data;

Element \*pointer;

};

Element\* CreateElement(int data){

Element \*e=new Element;

e->data=data;

e->pointer=NULL;

return e;

}

-----------------------------------------------------------------------------

LinkedList:

Null

head

tail

Data

pointer

Data

pointer

Data

pointer

Linked List L

InitLinkedList:

Null

head

tail

void initList(LinkedList& L){

L.head=L.tail=NULL;

}

-----------------------------------------------------------------------------

InsertFirst:

Null

Data

pointer

head

tail

Insert First

Null

head

tail

Null

head

tail

Data

pointer

Data

pointer

Data

pointer

Data

pointer

void InsertFirst(Element \*e, LinkedList &L){

if(L.head==NULL)

L.head=L.tail=e;

else{

e->pointer=L.head;

L.head=e;

}

}

-----------------------------------------------------------------------------

Insert Tail:

Insert tail

Case 1:

Null

head

tail

Case 2:

Null

head

tail

Data

pointer

Data

pointer

Data

pointer

Data

pointer

Null

Data

pointer

head

tail

-----------------------------------------------------------------------------

InsertAfterP:

tail

Data

pointer

Data

pointer

7

pointer

head

p

Travel List

100

pointer

2

Data

pointer

1

E

Null

void InsertAfterP(Element \*e, Element \*p){

if(p!=NULL){

e->pointer=p->pointer;

p->pointer=e;

}

}

Liên kết đôi:

Element

Null

Null

Data

pointer

pointer

Null

head

tail

Data

pointer

pointer

Data

pointer

pointer

Data

pointer

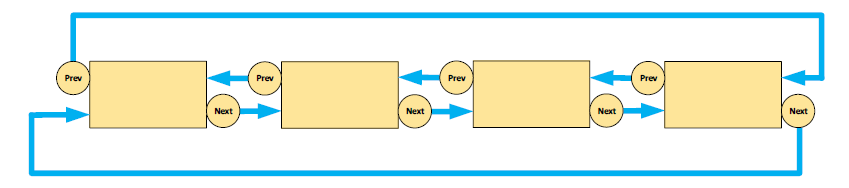
pointer

Linked List L

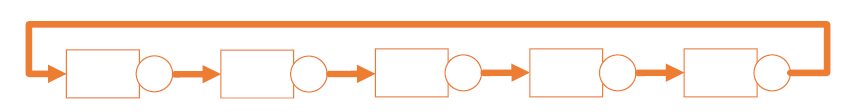
Null

Bài tập:

Liên kết vòng đôi:



Liên kết vòng đơn:



**Bài tập:**

1. Viết các hàm cho danh sách liên kết đôi, liên kết vòng đôi, liên kết vòng đơn:

+ thêm đầu, thêm đuôi.

+ Thêm sau 1 vị trí (phần tử p)

+ duyệt danh sách

+ xóa đầu, xóa đuôi, xóa tại 1 vị trí.

2. Sử dụng kiểu dữ liệu danh sách liên kết đơn, viết chương trình quản lý danh sách sinh viên, thông tin mỗi sinh viên bao gồm: mã số, họ tên, năm sinh, điểm trung bình. Thực hiện các yêu cầu sau:

*a) Khai báo, khởi tạo, và nhập danh sách.*

*b) Duyệt và in danh sách.*

*c) Tìm kiếm một sinh viên theo họ tên.*

*d) Chèn thêm một sinh viên mới vào sau một sinh viên có mã số chỉ định.*

*e) Xóa một sinh viên có mã sinh viên chỉ định.*

*f) Sắp xếp danh sách theo mã số sinh viên tăng dần.*

*g) Sắp xếp danh sách theo tên sinh viên theo thứ tự bảng chữ cái A, B, C,….*

*h) Hủy danh sách khi kết thúc chương trình.*

3. Cài đặt 1 DSLK đơn biểu diễn thông tin cho n thành phố. Thông tin các thành phố gồm: Tên, diện tích, dân số.

*a) Nhập/xuất dữ liệu cho DSLK.*

*b) Tìm thông tin thành phố có diện tích lớn nhất và thêm vào sau nó 1 thành phố.*

*c) Xóa khỏi DSLK thành phố “Ha Noi” nếu có.*

*d) Sắp xếp DSLK theo chiều tăng dần của dân số.*

4. Sử dụng kiểu dữ liệu danh sách liên kết đôi vòng, viết chương trình quản lý các chuyến bay của một công ty hàng không, mỗi chuyến bay gồm: Mã chuyến, ngày, giờ khởi hành, điểm đến. Thực hiện các yêu cầu sau:

- *Khai báo và khởi tạo danh sách list.*

- *Nhập danh sách bằng cách thêm vào list ở vị trí phù hợp để danh sách có thứ tự tăng của mã chuyến.*

- *Hãy in tất cả các chuyến bay khởi hành trong ngày chỉ định.*

- *Hủy danh sách khi kết thúc chương trình.*

5. Xây dựng danh sách liên kết với data là phân số( tử và mẫu số).

+ Viết hàm rút gọn tất cả các phân số trong danh sách.

+ Xóa các phân số có mẫu số là 1