#include<iostream>

using namespace std;

class Node {

public:

int data;

Node\* next;

Node(int val) {

data = val;

next = NULL;

}

};

class List{

Node\* head;

Node\* tail;

public:

List() {

head = tail = NULL;

}

void push\_front(int val) {

Node\* newNode = new Node(val);

if(head == NULL) {

head = tail = newNode;

return;

}

else {

newNode->next = head;

head = newNode;

}

}

void push\_back(int val) {

Node\* newNode = new Node(val);

if(head == NULL) {

head = tail = newNode;

return;

}

else {

tail->next = newNode;

tail = newNode;

}

}

void pop\_front() {

if(head == NULL) {

cout<<"List is EMPTY!!"<<endl;

return;

}

else {

Node\* temp = head;

head = head->next;

temp->next = NULL;

delete temp;

}

}

void pop\_back() {

if(head==NULL) {

cout<<"List is EMPTY!!"<<endl;

return;

}

else {

Node\* temp = head;

while(temp->next != tail) {

temp = temp->next;

}

temp->next = NULL;

delete tail;

tail = temp;

}

}

void insert(int val, int pos) {

if(pos<0) {

cout<<"INVALID Pos!!"<<endl;

return;

}

if(pos==0) {

push\_front(val);

return;

}

Node\* temp = head;

for(int i=0; i<pos-1; i++) {

if(temp==NULL) {

cout<<"INVALID Pos!!"<<endl;

return;

}

temp=temp->next;

}

Node\* newNode = new Node(val);

newNode->next = temp->next;

temp->next = newNode;

}

void del(int key) {

if(head==NULL) {

cout<<"List is empty"<<endl;

return;

}

else {

Node\* temp = head;

while(temp->next!=NULL) {

if(temp->next->data==key) {

head = head->next;

temp->next = NULL;

delete temp;

return;

}

}

}

}

int search(int key) {

Node\* temp = head;

int idx=0;

while(temp!=NULL) {

if(temp->data == key) {

return idx;

}

temp = temp->next;

idx++;

}

return -1;

}

void printll() {

Node\* temp = head;

while(temp != NULL) {

cout<<temp->data <<" -> ";

temp = temp->next;

}

cout<<"NULL"<<endl;

}

};

int main() {

List ll;

ll.push\_front(1);

ll.push\_front(2);

ll.push\_front(3);

// ll.push\_back(4);

ll.printll();

ll.insert(4,0);

ll.printll();

// ll.pop\_front();

// ll.printll();

// ll.pop\_back();

// ll.printll();

ll.del(2);

ll.printll();

cout<<"Pos is : "<<ll.search(3)<<endl;

return 0;

}

