package com.mycompany.main;

public class DoublyLinkerList {

node1 head;

node1 tail;

static int size=0;

public DoublyLinkerList(node1 head) {

this.head = head;

}

public DoublyLinkerList() {

}

public void add(int x){

node1 n=new node1(x);

if(size==0){

head=tail=n;

size++;

}

if(!search(x)){

if(size==1){

if(x>head.data){

tail.next=n;

n.back=tail;

tail=n;

size++;

}

else{

tail.back=n;

n.next=tail;

head=n;

size++;

}

}

else if(size>1){

if(head.data>x){

head.back=n;

n.next=head;

head=n;

size++;

}

else if(x>tail.data){

tail.next=n;

n.back=tail;

tail=n;

size++;

}

else{

node1 curr=head;

while(curr.next!=null){

if(x>curr.data&&x<curr.next.data){

n.next=curr.next;

curr.next.back=n;

curr.next=n;

n.back=curr;

size++;

break;

}

curr=curr.next;

}

}

}

}

}

public void add1(int x){

node1 n=new node1(x);

if(size==0){

head=tail=n;

size++;

}

if(size==1){

if(x>head.data){

tail.next=n;

n.back=tail;

tail=n;

size++;

}

else{

tail.back=n;

n.next=tail;

head=n;

size++;

}

}

else if(size>1){

if(head.data>x){

head.back=n;

n.next=head;

head=n;

size++;

}

else if(x>tail.data){

tail.next=n;

n.back=tail;

tail=n;

size++;

}

else{

node1 curr=head;

while(curr.next!=null){

if(x>curr.data&&x<curr.next.data){

n.next=curr.next;

curr.next.back=n;

curr.next=n;

n.back=curr;

size++;

break;

}

curr=curr.next;

}

}

}

}

public String print(){

node1 curr=head;

String s="";

for (int i = 0; i <size; i++) {

s=s+curr.data+" ";

curr=curr.next;

}

return s;

}

public void delete(int x){

if(head.data==x){

head=head.next;

size--;

}

else if(tail.data==x){

node1 b=tail.back;

b.next=null;

tail.back=null;

tail=b;

size--;

}

else{

node1 curr=head;

while(curr.next!=null){

if(curr.data==x){

curr.back.next=curr.next;

curr.next.back=curr.back;

size--;

break;

}

else{

curr=curr.next;

}

}

}

}

public boolean search(int x){

if(size==0){

return false;

}

else{

node1 curr1=head;

for(int i=0;i<size;i++){

if(curr1.data==x){

return true;

}

curr1=curr1.next;

}

}

return false;

}

public void update(int x,int y){

if(!search(y)){

node1 curr=head;

for(int i=0;i<size;i++){

if(curr.data==x){

delete(x);

add(y);//to print the elemnt in ascending order ;

}

else{

curr=curr.next;

}

}

}

}

}