There are two lines of electrical outlets, these outlets support different

voltages range from -100v to 100v. The volatges of the outlets in each line

are arranged in the ascending order as a singly linked lists, list1 and list2.

Your task is to combine these two lists into one list of outlets, and the

arrangement of outlets should be in the ascending order of their volatges.

NOTE: use singly linked list concept.

Sample Node:

============

class Node{

int data;

Node next;

public Node(int data) {

this.data = data;

this.next = null;

}

}

INPUT FORMAT:

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

Line-1: space separated integers, voltages of outlets in list-1

Line-2: space separated integers, voltages of outlets in list-2

OUTPUT FORMAT:

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

Space separated integers, voltages of outlets in ascending order.

SAMPLE INPUT-1:

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

1 3 5 7

2 4 6

SAMPLE OUTPUT-1:

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

1 2 3 4 5 6 7

SAMPLE INPUT-2:

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

-10 -5 -4 3 5 6

-6 -5 3 4 6 8

SAMPLE OUTPUT-2:

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

-10 -6 -5 -5 -4 3 3 4 5 6 6 8

My soln-but only got 75/100

import java.util.\*;

//Node class for reference.

/\*

class Node{

int data;

Node next;

public Node(int data) {

this.data = data;

this.next = null;

}

}

\*/

class Solution{

public static void add(Node l1,int val){

while(l1.next!=null){

l1=l1.next;

}

l1.next=new Node(val);

}

public Node mergeTwoLists(Node l1, Node l2){

// int l1,l2;

Node head=l1;

int count1=0;

int count2=0;

while(head!=null){

count1+=1;

head=head.next;

}

Node head1=l2;

while(head1!=null){

count2+=1;

head1=head1.next;

}

int i=0;

int j=0;

Node ll1=l1;

Node ll2=l2;

Node l=null;

while(i<count1 && j<count2){

if(ll1.data<=ll2.data){

if(l!=null){

add(l,ll1.data);

i+=1;

ll1=ll1.next;

}

else{

l=new Node(ll1.data);

i+=1;

ll1=ll1.next;

}

}

else{

if(l!=null){

add(l,ll2.data);

j+=1;

ll2=ll2.next;

}

else{

l=new Node(ll2.data);

j+=1;

ll1=ll1.next;

}

}

}

while(i<count1){

if(l==null){

l=new Node(ll1.data);

}

else{

add(l,ll1.data);

}

i+=1;

ll1=ll1.next;

}

while(j<count2){

if(l==null){

l=new Node(ll2.data);

}

else{

add(l,ll2.data);

}

j+=1;

ll2=ll2.next;

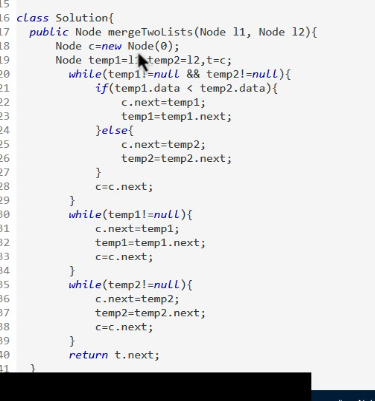
}

// System.out.println(count1);

return l;

}

}



Cliff Shaw is working on the singly linked list.

He is given a list of boxes arranged as singly linked list,

where each box is printed a positive number on it.

Your task is to help Mr Cliff to find the given list is equivalent to

the reverse of it or not. If yes, print "true", otherwise print "false"

Sample Node:

============

class Node{

int data;

Node next;

public Node(int data) {

this.data = data;

this.next = null;

}

}

Input Format:

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

Line-1: space separated integers, boxes as list.

Output Format:

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

Print a boolean a value.

Sample Input-1:

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

3 6 2 6 3

Sample Output-1:

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

true

Sample Input-2:

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

3 6 2 3 6

Sample Output-2:

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

False

Soln:

import java.util.\*;

/\*

// Node class for your reference.

class Node{

int data;

Node next;

public Node(int data) {

this.data = data;

this.next = null;

}

}

\*/

class Solution

{

boolean isPalindrome(Node head)

{

//Implement your logic here.

Node k=head;

String s="";

while(k!=null){

s+=String.valueOf(k.data);

k=k.next;

}

int start=0;

int end=s.length()-1;

while(start<=end){

if(s.charAt(start)!=s.charAt(end)){

return false;

}

start+=1;

end-=1;

}

return true;

}

}

'''

Cliff Shaw is working on the singly linked list.

He is given a list of boxes arranged as singly linked list, where each box is

printed with a positive number on it and arranged in the list are ascending order.

and numbers on the boxes may be repeated.

Mr Cliff Shaw is performing an operation on the list.

- To have only the distinct values in the final list,

removed all duplicate values.

Your task is to help Mr Cliff to perform the operation and return the updated list

Input Format:

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

Line-1: An integer, N number of boxes in list.

Line-2: N space separated integers, boxes as list.

Output Format:

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

Print the updated list.

Sample Input-1:

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

8

1 1 1 2 2 3 3 4

Sample Output-1:

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

1 2 3 4

Sample Input-2:

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

5

1 1 1 1 1

Sample Output-2:

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

1

Write your python code below using Slingly Linked list implementation.

'''