You are given the class HiHello that has four methods:

hi, hello, hihello and number.

The instance of HiHello will be passed to four different threads:

- Thread A: calls hi() that should output the word "Hi".

- Thread B: calls hello() that should output the word "Hello".

- Thread C: calls hihello() that should output the word "HiHello".

- Thread D: calls number() that should only output the integers.

The given class should output the series [1, 2, "Hi", 4, "Hello", ...]

where the i-th token (1-indexed) of the series is:

"HiHello", if i is divisible by 3 and 5,

"Hi", if i is divisible by 3 and not 5,

"Hello", if i is divisible by 5 and not 3, or

i, if i is not divisible by 3 or 5.

Your task is to Implement the HiHello class:

- HiHello(int n) Initializes the object with the number n that represents

the length of the sequence that should be printed.

- void hi(printHi) Calls printHi to output "Hi".

- void hello(printHello) Calls printHello to output "Hello".

- void hihello(printHiHello) Calls printHiHello to output "HiHello".

- void number(printNumber) Calls printnumber to output the numbers.

Input Format:

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

Line-1: An integer n.

Output Format:

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

Print the array as given in the sample.

Constraints:

• 1 <= n <= 10^4

Sample Input-1:

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

4

Sample Output-1:

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

1 2 Hi 4

Sample Input-2:

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

15

Sample Output-2:

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

1 2 Hi 4 Hello Hi 7 8 Hi Hello 11 Hi 13 14 HiHello  
  
Mr Rakesh is interested to work on Data Structures.

He has constructed a BinaryTree-BT.

He asked his friend Anil to check whether BT is self mirror tree or not.

Can you help Anil to find and return "true" if the given BT is a self mirror tree,

otherwise return "false".

Implement the class Solution:

1. public boolean isSelfMirrorTree(Node root):

- returns a boolean value.

NOTE:

- In the tree '-1', indicates empty(null).

Input Format:

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

A single line of space separated integers, values at the treenode

Output Format:

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

Print a boolean value.

Sample Input-1:

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

2 1 1 2 3 3 2

Sample Output-1:

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

true

Sample Input-2:

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

2 1 1 -1 3 -1 3

Sample Output-2:

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

False

/\*

//TreeNode Structure for Your Reference..

class Node{

public int data;

public Node left, right;

public Node(int data){

this.data = data;

left = null;

right = null;

}

}

\*/

class Solution {

public boolean isSelfMirrorTree(Node root) {

// Implement Your Code here..

return fun(root.left,root.right);

}

public boolean fun(Node l,Node r){

if(l==null && r==null){

return true;

}

if(l==null && r!=null){

return false;

}

if(l!=null && r==null){

return false;

}

if(l.data !=r.data){

return false;

}

return fun(l.left,r.right) && fun(l.right,r.left);

}

}

A set of computers connected with each other and formed a network with the

following rule, one computer can directly connected with atmost two computers only.

Each computer is assigned with an ID, the computer ID's may be repeated.

You are given the entire network as a tree.

Your task is to transform the network into a single row of computers,

The transformin process follows the below order:

- For transformation, use the same network tree structure.

- The transformation should be done in the pre-order format of the tree.

- In the network tree, the right computer points to the next computer

and the left computer connected to no computer.

Implement the class Solution:

1. public void transform(Node root) : transform the tree.

NOTE: in the input tree, consider -1 as 'No Connection'.

Input Format:

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

Single line of space separated integers, network ID's in the form the tree.

Output Format:

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

Print a string, as described in sample.

Sample Input-1:

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

1 2 3 4 5 6 7

Sample Output-1:

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

1 2 4 5 3 6 7

Explanation:

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

Look at the hint for unerstanding