Subtree of Another Tree

Try to solve the Subtree of Another Tree problem.

> **We'll cover the following** ︿
>
> - Statement
> - Examples
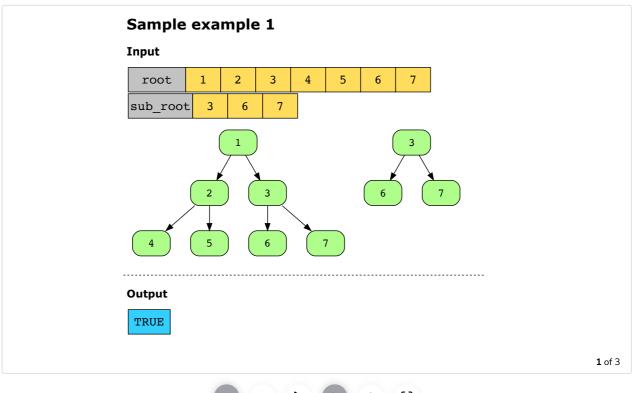> - Understand the problem
> - Try it yourself

Statement

Given the roots of two binary trees as `root` and `subRoot`, return TRUE if there is a subtree of `root` with the same structure and nodes of `subRoot`. Otherwise, return FALSE.

Constraints:

- The number of nodes in the `root` tree is in the range $[1, 2000]$.
- The number of nodes in the `subRoot` tree is in the range $[1, 1000]$.
- $-10^4 \leq$ `root.data` $\leq 10^4$
- $-10^4 \leq$ `subRoot.data` $\leq 10^4$

Examples

Sample example 1

Input

root	1	2	3	4	5	6	7

sub_root	3	6	7

Output
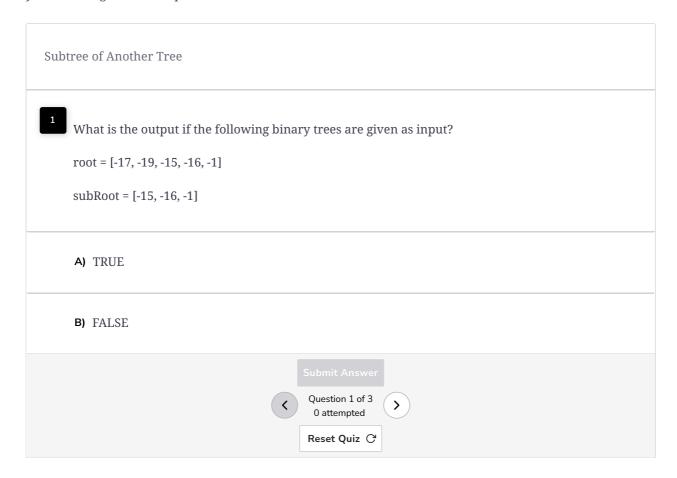
TRUE

Understand the problem

Let's take a moment to make sure you've correctly understood the problem. The quiz below helps you check if you're solving the correct problem:

Try it yourself

Implement your solution in `main.java` in the following coding playground.

> **Note:** The binary tree node's class has members `left` and `right` to store references to other nodes, along with the member `data` to hold the node's value.

Java

usercode > main.java

```java
1  import java.util.*;
2  import ds_v1.BinaryTree.TreeNode;
3
4  // Definiton of a binary tree node class
5  // class TreeNode<T> {
6  //     T data;
7  //     TreeNode<T> left;
8  //     TreeNode<T> right;
9
10 //     TreeNode(T data) {
11 //         this.data = data;
12 //         this.left = null;
13 //         this.right = null;
14 //     }
15 // }
16
```

```
19         // your code will replace this placeholder return statement
20         return false;
21     }
22 }
```

Test Cases   Results

| Case 1 | Case 2 | Case 3 |

Input #1

[100]

Input #2

[100]

Subtree of Another Tree

🔆 Hide Hint

You might want to go over the Tree Depth-first Search pattern again.

✅ Mark as Completed