Langkah Awal Competitive Programming

Maulana Ifandika

Same Tree

**Soal**

Diberikan 2 tree, tulis fungsi untuk cek apakah tree tersebut sama dari segi struktur dan nilai.

**Ketentuan**

- Jika panjang berbeda maka false

- Jika nilai berbeda TreeNode a dan b posisis sama maka false

**Solusi**

- Kita jelajahi TreeNode kiri lalu kanan dan cek setiap nilai TreeNode kiri dan kanan.

- Jika kedua node kiri dan kanan sama2 kosong maka panjang sama maka return true.

- Jika salah satu TreeNode kosong maka panjang tidak sama, contoh TreeNode kiri sudah kosong tapi kanan masih lanjut maka stukturnya tidak sama ada yang lebih, berati return false.

- Lalu ulangi dengan cek setiap nilai dari masing2 TreeNode jika nilai TreeNode kiri == TreeNode kanan lalu ulangi untuk l.kiri dan r.kiri dan juga l.kanan dan r.kanan.

class TreeNode {

int val;

TreeNode left;

TreeNode right;

TreeNode() {}

TreeNode(int val) { this.val = val; }

TreeNode(int val, TreeNode left, TreeNode right) {

this.val = val;

this.left = left;

this.right = right;

}

}

public class SameTree {

public static boolean isValid(TreeNode l, TreeNode r) {

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

return true;

}

else if(l == null || r == null) {

return false;

}

return (l.val == r.val && isValid(l.left, r.left) && isValid(l.right, r.right));

}

}

Bilangan Prima

**Soal**

“Bilangan prima adalah bilangan bulat > 1 dan memiliki 2 faktor, 1 dan bilangan itu sendiri”

Tentukan apakah nilai n adalah bilangan prima atau bukan.

**Sulusi**

- Cek jikai nilai n < 2 maka bukan bilangan prima.

- Lakukan perulangan dari angka 2 tidak dari 1 karena bilangan prima > 1.

- Jika n modulo (%) 2,3,4,5,… maka return false karena memiliki lebih dari 2 faktor.

public class PrimeNumber {

public boolean solution(int n) {

if(n < 2) return false;

for(int i = 2; I <= n; i++) {

if(n % I == 0) {

return false;

}

}

Return true;

}

Generating List Prime Number

**Soal**

Buat daftar bilangan prima.

**Solusi**