**Binary Trees - Mirroring**

class Node{

    constructor(val){

        this.val = val;

        this.left = null;

        this.right = null;

    }

}

const two = new Node(2);

const three = new Node(3);

const four = new Node(4);

const five = new Node(5);

const six = new Node(6);

five.right = six;

five.left = three;

three.left = two;

three.right = four;

const inorderTraversal = (root) => {

    if(!root) return []

    return [... inorderTraversal(root.left), root.val, ...inorderTraversal(root.right)]

}

console.log(inorderTraversal(five));

function mirrorBT(root) {

    if (root == null)

    {

        return null;

    }

    let mirror = new Node(root.val)

    mirror.right = mirrorBT(root.left)

    mirror.left = mirrorBT(root.right)

    return mirror;

}

console.log(inorderTraversal(mirrorBT(five)));