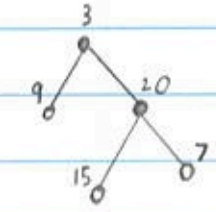
Find Max Depth of Binary Tree

```
type TreeNode struct {
        Val   int
        Left  *Tree Node
        Right *Tree Node
}
```

```
func maxDepth (root *Tree Node) int {
    if root == nil {
            return 0
    } else if root.Left == nil && root.Right==nil {
            return 1
    } else {
        var l int = maxDepth (root. Left)
        var r int = maxDepth ( root. Right )
        if l > r {
                return 1 + l
        } else {
                return 1 + r
        }
    }
```

Binary Tree: [3 , 9 , 20 , null , null , 15 , 7] ,



0) max Depth (root. Left) ↦ l:= 1

1) max Depth (root. Right) ↦ maxDepth (root. Left) ↦ l:=1

max Depth (root. Right) ↦ r:=1

since l not > r , return 1+r

2) return l+r = 1+1 = $\boxed{2}$