

Algorithm Height (T)

Return heightHelper(T,T.root()). $O(1)$

Algorithm heightHelper(T,V)

If(T.isExternal(V)) return 0; $O(1)$

Else

leftH:=1+heightHelper(T,T.leftChild()). $O(n)$

rightH:= 1+heightHelper(T,T.rightChild()) $O(n)$

if(leftH>rightH) return leftH; $O(1)$

else return rightH; $O(1)$

over all time complexity is $O(n)$

```
function height(T){
    return heightHelper(T,T.root())
}
function heightHelper(T,V){
    if(T.isExternal(V)) return 0
    else{
        let leftH=1+heightHelper(T,T.leftChild(V))
        let rightH=1+heightHelper(T,T.rightChild(V))
        if(leftH>rightH)
            return leftH;
        return rightH;
    }
}
```