Algorithms on trees: Depth

- Depth:
  - depth(T, v) is the number of ancestors of v, excluding v itself
- Recursive formulation
  - if v == root, then depth(v) = 0
  - else, depth(v) is 1 + depth (parent(v))
- Compute the depth of a node v in tree T: int depth(T, v)
- Algorithm:

```
int depth(T,v) {
     if T.isRoot(v) return 0
     return 1 + depth(T, T.parent(v))
}
```

- Analysis:
  - O(number of ancestors) = O(depth_v)
  - in the worst case the path is a linked-list and v is the leaf
  - ==> O(n), where n is the number of nodes in the tree