Exercise 6 - Solutions

1. Recall definition A:

$$\text{depth}(v) = \begin{cases} 0 & \text{if } v \text{ is root of } T \\ 1 + \text{depth}(p) & \text{where } p \text{ is parent of } v \text{ otherwise} \end{cases}$$

$$\text{height}_A(T) = \max\{\text{depth}(v) : v \text{ is leaf of } T\} \qquad (*)$$

Definition B:

$$\text{height}(v) = \begin{cases} 0 & \text{if } v \text{ is leaf} \\ 1 + \max\{\text{height}(c) : c \text{ is child of } v\} & \text{otherwise} \end{cases} \qquad (*)$$

$$\text{height}_B(T) = \text{height}(r) \quad \text{where } r \text{ is root of } T$$

We first prove that $\text{height}_A(T) \leq \text{height}_B(T)$:

Let l be the leaf where the max in $(*)$ is taken. So $d := \text{depth}(l) = \text{height}_A(T)$ and there is a path $r, v_1, v_2, \dots, v_d = l$ of length d from r to l.

Then:

$$\text{height}(r) \geq 1 + \text{height}(v_1) \geq 2 + \text{height}(v_2) \geq \dots \geq d + \underbrace{\text{height}(l)}_{=0} = \text{height}_A(T)$$

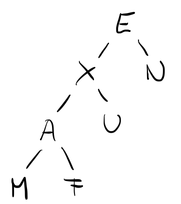
Next, we prove that $\text{height}_A(T) \geq \text{height}_B(T)$:

Let c_i be the child at which the max in $(**)$ is taken at level i when computing $h = \text{height}(r)$. So i ranges from $1, \dots, h$ and c_h is a leaf.

$$\Rightarrow \quad \text{height}(r) = \text{depth}(c_h) \leq \max\{\text{depth}(l) : l \text{ is a leaf}\} = \text{depth}_A(T).$$

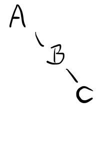
2+3: See Python code on class page

4.



5. (i) In pre-order traversal, root is always first
In post-order traversal, root is always last
$\Rightarrow$ pre-order $\neq$ post-order if tree has more than one node.

(ii) If all nodes have only left or only right children, e.g.



has preorder A-B-C, post-order C-B-A.