Data Structures (2026 Fall)
Assignment 03

Chuang-Chieh Lin

Department of Computer Science and Engineering,
National Taiwan Ocean University

1. Rewrite the postorder tree traversal algorithm in the lecture to be an iterative one.

2. Each node of a binary tree has fields $key(x)$, $left(x)$, $right(x)$, and $parent(x)$ (where $parent(\text{root}) = \texttt{NULL}$). Assume all keys in the tree are distinct. Write an algorithm MoveToRoot(x, k) that takes as input a pointer x to the *root* of a binary tree and a key k, and behaves as follows:
   (a) If there is no node v in the tree with $key(v) = k$, then the algorithm
      i. must not modify the tree, and
      ii. must return the original root x.
   (b) If there is a node v in the tree with $key(v) = k$, then the algorithm must:
      i. find such a node v;
      ii. repeatedly *swap* the key values on the unique path from v up to the root, that is, while $parent(v) \neq \texttt{NULL}$, exchange $key(v)$ and $key(parent(v))$, and then set $v \leftarrow parent(v)$;
      iii. after these swaps, ensure that the root node has key k, while the shape of the tree and all pointer fields $left(\cdot)$, $right(\cdot)$, and $parent(\cdot)$ remain unchanged;
      iv. return the root node x.

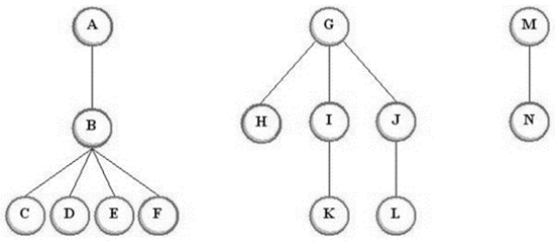
   Your algorithm is allowed to read and write the $key(\cdot)$ fields of nodes but may not change any $left(\cdot)$, $right(\cdot)$, or $parent(\cdot)$ pointers and may not create or delete nodes. Use clear pseudocode.

3. Consider the following array:

$$A = \langle 4, 33, 6, 90, 33, 32, 31, 91, 90, 89, 50, 33 \rangle.$$

   (a) Is A a min-heap? Justify your answer by briefly explaining the min-heap property.
   (b) Please write an $O(n)$ time algorithm which checks whether the input binary tree (stored in an array) is a min-heap. Justify your answer and analyze the time complexity.

4. Transforming the forest into a binary tree using the rule:
   - If $T_1, T_2, \ldots, T_n$ is a forest of trees, then the binary tree corresponding to this forest is denoted by $B(T_1, T_2, \ldots, T_n)$.
   - The root of this binary tree is root(T_1).
   - Its left subtree is $B(T_{11}, T_{12}, \ldots, T_{1m})$, where $T_{11}, T_{12}, \ldots, T_{1m}$ are the subtrees of root(T_1); and its right subtree is $B(T_2, T_3, \ldots, T_n)$.



5. Prove that $\sum_{h=1}^{\infty} \frac{h}{2^h} = O(1)$.