University of Houston

COSC 3320: Algorithms and Data Structures
Spring 2017

Homework 4

Due March 9, at the start of class

1. Given an array $A[1, 2, \ldots, n]$ of n elements, a *majority* element of A is an element occurring at least $\lceil (n+1)/2 \rceil$ times. The elements cannot be ordered or sorted, but can be compared for equality. Design an efficient divide and conquer algorithm that returns a majority element of A (if any), and determine its complexity.

2. Consider the output A of the following algorithm.

```
w1 = 'blacksmith'
w2 = 'powdery'
A = new array of size 10
B = new array of size 7
for i = 0 to 9 do
   A[i] = (i+1)-th letter of w1
for i = 0 to 6 do
   B[i] = (i+1)-th letter of w2
   j = (i+1)-th digit of your 7-digit UH ID
   A[j] = B[i]
return A
```

   You are told that $\sigma = A[0], A[1], \ldots, A[9]$ is the sequence of nodes of a tree visited by some visit procedure.

   (a) Exhibit one tree T whose nodes are labeled with the letters in A and whose preorder visit would visit the nodes of T in the order given by σ.

   (b) Exhibit one tree T whose nodes are labeled with the letters in A and whose postorder visit would visit the nodes of T in the order given by σ.

   (c) Exhibit one binary tree T whose nodes are labeled with the letters in A and whose inorder visit would visit the nodes of T in the order given by σ.

3. Let T be a proper binary tree. Given a node $v \in T$, the *imbalance* of v, denoted imbalance(v), is defined as the difference, in absolute value, between the number of leaves of the left subtree of v and the number of leaves of the right subtree of v. (If v is a leaf, then imbalance(v) is defined to be 0.) Define imbalance(T) $= \max_{v \in T}$ imbalance(v).

   (a) Provide an upper bound to the imbalance of a proper binary tree with n nodes, and exhibit a proper binary tree whose imbalance matches such an upper bound.

   (b) Draw a proper binary tree T for which imbalance(T) $=$ imbalance(v) and v is *not* the root of T.

(c) Design an efficient algorithm to compute imbalance(T), and determine its complexity.

4. A proper d-ary tree, with $d \geq 1$, is a tree where every node has either 0 or d children.

   (a) Draw two proper ternary ($d = 3$) trees T_1 and T_2 with 8 internal nodes each and such that T_1 has the maximum possible height and T_2 has the minimum possible height. How many leaves do T_1 and T_2 have?

   (b) Let T be a proper d-ary tree with n nodes, m of which are leaves, and of height h. Which one of the following relations can be true for any T?

      i. $m = (d-1)^h + 1$

      ii. $m = d(n - m - 1) - 1$

      iii. $m = (d-1)(n-m) + 1$

   (c) Prove the relation guessed in (b).