CS 263
Data Structures
ASSIGNMENT # 2
DUE DATE: Tuesday, October 22, 2013

If you are working in a group of 2 or three, please submit one copy with all of your names and student numbers on each sheet. Please use a fresh sheet of paper for each question.

1. Question 2 from Homework 1. (I gave you an extension on this question.)

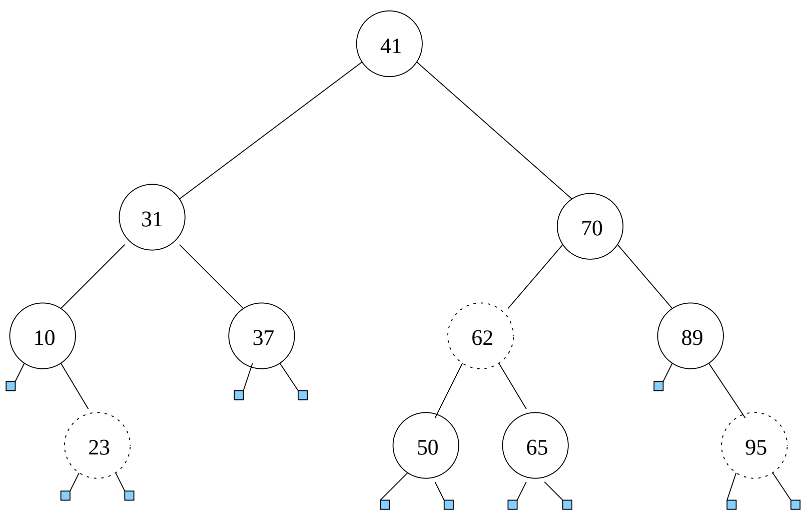
2. Suppose 3 values A, B, and C are chosen uniformly and independently from the set of integers $\{1, \ldots, r\}$, where $r \geq 1$.

   (a) What is the probability that all three values are the same? Briefly justify your answer.

   (b) What is the probability that all three values are different? Briefly justify your answer.

   (c) What is the expected number of different values? Briefly justify your answer.

3. Consider the following binary search tree T.



   Solid nodes are black, dotted nodes are red.

   (a) Draw the red-black tree that results from inserting the key 15 into T.

   (b) Draw the red-black tree that results from deleting the key 37 from the original tree T.

4. Consider a binary tree T. Let $|T|$ be the number of nodes in T. Let x be a node in T, let L_x be the left subtree of x and let R_x be the right subtree of x. We say that x has the "approximately balanced property", $ABP(x)$, if $|R_x| \leq 2|L_x|$ and $|L_x| \leq 2|R_x|$.

   (a) What is the maximum height of a binary tree T on n nodes where $ABP(root)$ holds? Justify your answer.

(b) We call T an ABP-tree if $ABP(x)$ holds for every node x in T. Prove that if T is an ABP-tree, then the height of T is $O(\log n)$. More precisely, show that

$$height(T) \leq \log_2 n / \log_2 \frac{3}{2}$$

.

5. Suppose we are given a bit-vector $A = A[1] \ldots A[n]$ of length n (where $A[i]$ is either 0 or 1). We wish to determine if at least half the elements in A are 1's. Consider the following algorithm:

```
HALFONES( A )
    numOnes ← 0
    numZeros ← 0
    for i = 1 to n do
        if A[i] = 1 then
            numOnes + +
            if numOnes ≥ n/2 then return true
        else
            numZeros + +
            if numZeros > n/2 then return false
```

Measure the complexity by counting the number of array comparisons performed.

(a) What is the best case complexity of HALFONES? Do not use asymptotic notation. Justify your answer.

(b) What is the worst case complexity of HALFONES? Do not use asymptotic notation. Justify your answer.

(c) What is the average case complexity of HALFONES, assuming a uniform distribution? Do not use asymptotic notation. Justify your answer. You may express your answer as a sum.

Remember to formally define the sample space, the probability distribution function, and any necessary random variables, as described in class. You do not need to mathematically simplify your answer.

6. We want to augment Red-Black Trees to support the following query, AVERAGE(x), which returns the average key-value in the subtree rooted at node x (including x itself). The query should work in worst-case time $\Theta(1)$.

(a) What extra information needs to be stored at each node?

(b) Describe how to modify INSERT to maintain this information, so that its worst-case running time is still $O(\log n)$. Briefly justify your answer.

(c) Describe how to modify DELETE to maintain this information, so that its worst-case running time is still $O(\log n)$. Briefly justify your answer.

You may find it helpful to implement Red-Black trees using the code from the text, and then modify your code to produce an augmented tree for this problem.