CS210a Data Structures and Algorithms
Fall 2013
Assignment 3
Due date: October 23
Total of 20 Marks

Put your assignment in an envelope labelled with your name and course number and drop it in the CS2210 locker (locker #302 located on the third floor of the Middlesex College Building) by midnight on the due date. You need to include a signed submission form.

You might find this fact useful: $\sum_{i=1}^{n-1} i = \frac{n(n-1)}{2}$.

1. (2 marks) Consider a hash table of size $N = 7$ where we are going to store integer values. The hash function is $h(k) = k \bmod 7$. Draw the table that results after inserting, in the given order, the following values: 19, 27, 12, 47, 15. Assume that collisions are handled by separate chaining.

2. (2 marks) Show the result of the previous exercise, assuming collisions are handled by linear probing.

3. (2 marks) Repeat exercise (1) assuming collisions are handled by double hashing, using a secondary hash function $h'(k) = 5 - (k \bmod 5)$.

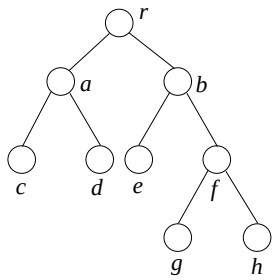
4. (4 marks) Solve the following recurrence equation **and** give the order of $f(n)$. You must show how you solved the equation.

   $f(1) = 1$

   $f(n) = f(n-1) + 2(n-1)$

5. - (7 marks)Write an algorithm `count(r,k)` that receives as input the root r of a proper binary tree T and a positive integer value k and it outputs the number of leaves of T that are at distance k from r.

     For example, for the following tree and $k = 2$, the algorithm must output the value 3 as there are 3 leaves c, d, e at distance 2 from r. For the following tree and $k = 4$ the algorithm must return the value 0 as there are no leaves at distance 4 from r.



   - (3 marks) Compute the worst case time complexity of your algorithm as a function of the total number of nodes in the tree. You must give the order of the time complexity of the algorithm, and you must explain how you computed it.