CS210a Data Structures and Algorithms
Assignment 3
Due date: October 28
Total of 20 Marks

Put your assignment in a $9'' \times 12''$ envelope labelled with your name and course number and drop it in the CS2210 locker (locker #300 located on the third floor of the Middlesex College Building) by 11:59 pm 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: 5, 15, 12, 26, 11. 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 secondary hash function $h'(k) = 5 - (k \bmod 5)$.

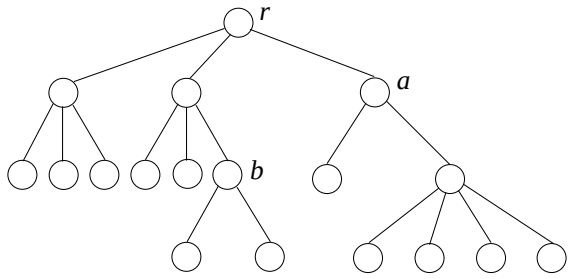
4. (3.5 marks) Solve the following recurrence equation **and** give the order of $f(n)$. You must show how you solved the equation.

   $f(1) = 3$

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

5.(i) (7 marks)Write in pseudocode an algorithm `min-degree(r)` that receives as input the root r of a tree and it outputs the minimum degree of the **internal** nodes in the tree.

For example, for the following tree the algorithm must output the value 2 as node a is an internal node with minimum degree and its degree is 2. Note that node b also has minimum degree but this fact does not change the output that the algorithm must produce.



5.(ii) (3.5 marks) Compute the worst case time complexity of your algorithm as a function of the total number n 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.