Due: Thu Apr 25th at the start of class

Homework #4
CMSC351 - Spring 2019

PRINT Name: ___________________________

- Grades depend on neatness and clarity.
- Write your answers with enough detail about your approach and concepts used, so that the grader will be able to understand it easily. You should ALWAYS prove the correctness of your algorithms either directly or by referring to a proof in the book.
- Write your answers in the spaces provided. If needed, attach other pages.
- The grades would be out of 16. Four problems would be selected and everyone’s grade would be based only on those problems. You will also get 4 bonus points for trying to solve all problems.

1. Give short answers to these questions:
   - Sum of degrees of a connected graph with n vertices is at least ________
   - What is the number of edges of a complete bipartite graph whose vertices is partitioned into two independent sets $V_1$ and $V_2$? (based on $|V_1|$ and $|V_2|$) ________________
   - What is the minimum number of vertices of a graph with 20 edges? __________
   - In which dynamic data structures we can do insert and delete in average O(1) time? ________
2. Given two sets $S_1$ and $S_2$, and a real number $x$, find whether there exists an element from $S_1$ and an element from $S_2$ whose sum is exactly $x$. The algorithm should run in average time $O(n)$, where $n$ is the total number of elements in both sets. (Try to use the data structures you have learned in the lectures.)
3. A proper coloring of a graph is a way of coloring the vertices of a graph such that no two adjacent vertices have the same color. For a graph $G$, $\Delta(G)$ denotes the maximum degree of a vertex in $G$. Design an algorithm that given a graph $G = (V, E)$, finds a proper coloring of $G$ with at most $\Delta(G) + 1$ colors in time $O(|V| + |E|)$. (Hint: Use the adjacency list representation of $G$).
4. [Prob 4.16, Pg 88] Design a data structure that supports the following operations. Each operation should take $O(\lg n)$ time in the worst case where $n$ is the number of elements in the data structure. Explain the required process for doing each operation and prove that it takes $O(\lg n)$ time.

   a) $\text{Insert}(x)$: Insert the key $x$ into data structure only if it is not already there.

   b) $\text{Delete}(x)$: delete the key $x$, if it is there!

   c) $\text{Find\_Smallest}(k)$: find the $k$th smallest key in the data structure.
5.  [Prob 4.22,Pg 88]
   a) Determine the general structure of a simple binary search tree formed by inserting the numbers 1 to \( n \) in order (you may draw a general structure, but explain it properly). What is the height of this tree?

   b) Determine the general structure of an AVL search tree formed by inserting the numbers 1 to \( n \) in order. What is the height of this tree?
6. [Prob 6.34,Pg 179] The input is a heap of size \( n \) (in which the largest element is on top), given as an array, and a real number \( x \). Design an algorithm to determine whether the \( k \)th largest element in the heap is less than or equal to \( x \). The worst-case running time of your algorithm should be \( O(k) \), independent of the size of the heap. You can use \( O(k) \) space. (Notice that you do not have to find the \( k \)th largest element; you need only to determine its relationship to \( x \).)
7. By using the concepts of graph theory, prove that removing opposite corner squares from an $8 \times 8$ checkerboard leaves a sub-board that cannot be partitioned into $1 \times 2$ and $2 \times 1$ rectangles.
8. As we defined in the class, a bipartite graph is a graph whose vertices V can be partitioned into two sets V1 and V2 such that there is no edge between two vertices in V1 and there is no edge between two vertices in V2. Using the definition above prove that a bipartite graph cannot have a simple cycle with an odd number of vertices.
9. [hard] We have a big rectangle room which is tiled by a finite number of non-overlapping rectangles. The sides of all the tiles are parallel to the sides of the room. We know that the length of at least one side of each tile is integer. Prove that at least one side of the room is also integer. (Hint: The sum of the degrees is always even in a graph!)