CSE 565	Martin Fürer	Algorithm Design and Analysis

Assignment 3 Given: 9/5/18 Due: Wednesday 9/12/18

Exercises

Exercises are for your own practice. Don't hand them in.

1. Solve Exercise 5, Chapter 3 on page 108 of the Textbook.
   (Number of nodes in a binary tree)

2. Solve Exercise 6, Chapter 3 on page 108 of the Textbook.
   (Same tree for BFS and DFS)

Problems

Problem solutions have to be handed in. A subset of them will be graded.

Remember: Even though collaboration is permitted, you have to write the solutions by yourself without assistance. Be ready to explain them orally if asked. Write your collaborators on your first sheet. You are not allowed to get solutions from outside sources such as the Web or students not enrolled in this class.
For additional information see the Course Announcement on Canvas.

1. [10+10=20 points] (Sum of functions)
   Let $f_1, f_2, \ldots$ be an infinite sequence of positive functions such that for all $i \geq 1$, we have $f_i(n) = O(n)$.

   (a) For any fixed positive integer m, define $s_m(n) = \sum_{i=1}^{m} f_i(n)$, and prove a good upper bound on $s_m(n)$ in O-notation.

   (b) For $t(n) = s_n(n) = \sum_{i=1}^{n} f_i(n)$, prove or disprove: $t(n) = O(n^2)$.

2. [10+10=20 points] (Shortest Cycles)
   Assume, we want to find a shortest directed cycle C in directed graph $G = (V, E)$. To do this, we employ a BFS (breadth-first search) algorithm. The BFS algorithm computes the depth of every vertex in a BFS tree.

(a) First we look at structural properties of such a shortest cycle C in the BFS tree. Let $C = (v_1, v_2, \ldots, v_k, v_1)$ for some $k \geq 2$. (If (u, v) and (v, u) are edges, then (u, v, u) is considered to be a directed cycle of length 2.)

Assume v_i (for some i with $1 \leq i \leq k$) is the starting point of a BFS, i.e., v_i is at depth 0. At which depth are the other vertices of C? Why?

(b) Describe (with text or pseudo-code) an algorithm to find a shortest directed cycle in G. Your algorithm should run in time $O(n(m + n))$, where $m = |E|$ and $n = |V|$.

3. [20+5 = 25 points] (Biconnected components)

An undirected graph representing an electricity grid should really have the property of being biconnected. This has the effect that if a tree falls on one power line, all the lights are still on. An undirected graph is biconnected, if it is connected and every edge belongs to a simple cycle.

(a) Design an DFS-based algorithm A that decides whether a graph G is biconnected. Write pseudo-code for A. Use a recursive procedure to describe A.

Instead of just identifying which vertices are already visited, your DFS algorithm should compute the depth of each vertex in the DFS tree. Before a vertex is placed in the tree, its depth is initialized to ∞.

Hint: Each vertex v has to learn the depth $v.h$ of the highest vertex in the DFS tree T to which there is an edge from some vertex in the subtree of T rooted at v.

(b) Every vertex v knows an increasingly better upper bound on $v.h$. At which time does v know the proper value of $v.h$?