CSE 565	**Martin Fürer**	**Algorithm Design and Analysis**

Assignment 4 Given: 9/18/18 Due: Monday 9/24/18

Exercises

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

1. Solve Exercise 1, Chapter 4 on page 188 of the Textbook.
   (Is the lightest edge in an MST?)
2. Solve Exercise 5, Chapter 4 on page 190 of the Textbook.
   (Communication tower placement)
3. Solve Exercise 8, Chapter 4 on page 192 of the Textbook.
   (MST with distinct edge weights)

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+ 5+10=25 points] (Shortest distances in a DAG)
   Your input is a DAG (directed acyclic graph) $G = (V, E)$ with a function $\ell : E \to \mathbb{N}$ defining a length for each edge. Furthermore, a vertex $s \in V$ with no edges into s is given. To compute the shortest distances from s to all vertices, one could use Dijkstra's algorithm.
   (a) Find and describe a more efficient algorithm, using the fact that G is a DAG. You may call another well known algorithm, just saying what it does without describing how it does it.
   (b) What is the running time of your algorithm as a function of $n = |V|$ and $m = |E|$?
   (c) Argue that your algorithm is correct. Hint: Your argument might be very similar to that for Dijkstra's algorithm.

2. [5+5+5+5=20 points] (d-ary heaps)
   A d-ary heap is defined analogously to a binary heap (see Chapter 2.5 of our textbook), except that with at most one exception, internal nodes have d instead of 2 children.

   (a) Consider an array implementation of d-ary heaps. Note that for $d > 2$ it is most convenient to put the root in position 0 of the array (the index of the root is 0), while for $d = 2$ it is better to put the root at position 1. How can you compute Child(i, k), the index of the k-th child of the node with index i? How can you compute Parent(i), the index of the parent of the node with index i?

   (b) What is the minimal number of elements of a d-ary heap of height h? Note that the eight of a tree is measured with edges. E.g., a 1-vertex tree has height 0.

   (c) On Insertion of a new element into a d-ary heap with n nodes, we need to run a procedure Heapify-up as on page 61 of the textbook. Write the procedure for d-ary heaps, and analyze its running time as a function of n and d. Θ-notation is good enough, but remember that d is not considered to be a constant.

   (d) ExtractMin requires a procedure Heapify-down as on page 63 of the textbook. Analyze the time of Heapify-down for a d-ary heap, again as a function of n and d.

3. [10+10=20 points] Solve Problem 2, Chapter 4 on page 189 of the Textbook. (Modified lengths for MST and Shortest Paths)

4. [10 points] Solve Problem 3, Chapter 4 on page 189 of the Textbook. (Greedily filling each truck is good)