Data Structures and Algorithms
FR 6.2 Informatik

Sanders, Telikapalli WS 03/04

http://www.mpi-sb.mpg.de/~sanders/courses/algdat03/index.html

---

Up to **four** *people can work on an exercise together. But each of you should be able to explain the solutions to the TA (Bremser). Write your names* **and** *the name of your group (time, TA) on the sheets. Staple them together*

Assignment 2

Deadline: November 10, 2003

Solve at least two of the following four exercises completely (or more of them partially)

Exercise 1

Suppose you want to solve the SSSP on a graph with positive edge weights. Let

$$r = \frac{\max_{e \in E} c(e)}{\min_{e \in E} c(e)}.$$

Develop an algorithm that runs in time $\mathcal{O}(m + nr)$ for such inputs. Hint: Use a bucket queue with buckets of width $\min_{e \in E} c(e)$. Show that *all* vertices in the smallest nonempty bucket have $d(v) = \mu(s, v)$. Describe your algorithm design and prove the claimed runtime bound.

Exercise 2

Design a family of graphs and a non-negative cost function such that the relaxation of $m-(n-1)$ edges causes a *decreaseKey* operation for the Dijkstra algorithm.

Exercise 3

Running in Saarbrücken

To get in shape, you have decided to start running to the university. You want a route that goes entirely uphill and then downhill so that you can work up a sweat going uphill and then get a nice breeze at the end of your run as you run faster downhill. Your run will start at home and end at the university and you have a map detailing the roads with m road segments (any existing road between two intersections) and n intersections. Each road segment has a positive length, and each intersection has a distinct elevation.

1. Assuming that every road segment is either uphill or downhill, give an efficient algorithm to find the shortest route that meets you specifications.
2. Give an efficient algorithm to solve the problem if some roads may be level (i.e., both intersections at the end of the road segments are at the same elevation) and therefore can be taken at any point.

Exercise 4

Give an n-element set of K-bit integers such that the veb search tree takes space $\Omega(n \log K)$.