Chip Design
SS 2008

Prof. Dr. Jens Vygen
Stephan Held

Exercise 8

1. Modify Dijkstra's algorithm to prove the following statements.

   (a) Given a digraph $G = (V, E)$, $s \in V$, and a length function $l : E \to \mathbb{Z}_+$, and an upper bound Δ on $\max\{dist_l(s, v) \mid v \text{ reachable from } s\}$, a shortest path tree rooted at s can be found in time $O(m + \Delta)$. (3 points)

   (b) Given a digraph $G = (V, E)$, $s \in V$, and a length function $l : E \to \mathbb{Z}_+$, a shortest path tree rooted at s can be found in time $O(m \log_d L)$, where $d = \max\{2, m/n\}$, and $L := \max\{l(e) \mid e \in E\}$. (5 points)

2. Implement the fractional packing algorithm to pack two-point connections in a uniform (two-dimensional) grid graph. The first line in the input file contains three integer tokens defining the graph: n, m, and k. Here the first token n is the number of rows in the graph, m is the number of columns and k is the edge capacity applying equally to all edges. Thus all vertices can be specified by two indices (i, j) for their row $0 \le i < n$ and column $0 \le j < m$ positions. Let $(0, 0)$ refer to the vertex in the lower left corner of the grid graph. The indices are also interpreted as geometric positions. Thus, all edge lengths are 1. Now each line following the graph definition specifies a pair of vertices which have to be connected. Example:

   ```
   1 2 3 4
   ```

   specifies that vertex $(1, 2)$ is to be connected with $(3, 4)$. Dijkstra's algorithm may be implemented with quadratic running time (without using a heap). The output should be a single number specifying the maximum congestion.

   (12 points)

The deadline for Problem 1. is **Tuesday June 17 at 12:15**, before the lecture.

The deadline for Problem 2. is **Tuesday July 1 at 12:15**, before the lecture.