Exercise 1 (3 points)

Consider an arbitrary tree $T$ and a set $W$ of vertices in $T$, where $W$ is of even size. Give a distributed algorithm that computes a route-disjoint matching of the vertices in $W$ in time $O(D(T))$.

Exercise 2 (2+3 points)

The following policy is proposed for solving the “smallest $k$ of $m$” problem described in section 4.3.1 of the Peleg book:

- At any given moment along the execution, every vertex keeps only the set of (at most) $k$ smallest items it knows.
- In each step, each vertex sends to its parent an arbitrary element from this set that hasn’t been sent yet.

a) Prove that using this policy, the $k$ smallest elements must still arrive at the root by time $O(k \cdot D(T))$.

b) For every integer $m \geq 1$, give an example of a tree $T$ and an initial distribution of $m$ elements, where $k = \sqrt[3]{m}$ and the above process takes $\Omega(m)$ steps.

Exercise 3 (3+6 points)

Consider Dijkstra’s algorithm from the lecture.

a) Prove the tightness of the message complexity analysis of Dijkstra’s algorithm by establishing the following (existential) lower bound: For arbitrary integers $n$ and $1 \leq D \leq n-1$, there exists an $n$-vertex $D$-diameter graph $G = (V, E)$ with $|E| = O(n)$ on which the execution of Dijkstra’s algorithm requires $\Omega(nD + |E|)$ messages.

b) Prove or disprove the following lower bounds:

For every $n$-vertex, $D$-diameter graph $G = (V, E)$, Dijkstra’s algorithm requires at least

1. $\Omega(nD)$ messages
2. $\Omega(|E|)$ messages
3. $\Omega(D^2)$ time.

Exercise 4 (5 points)

Modify the Bellman-Ford algorithm so that it detects its termination. Make sure that the given bounds on the time and message complexity still hold.