Exercises for Unit 24

1. Consider the Jarnik-Prim algorithm for computing a minimum spanning tree. Convince yourself that it can be implemented with Fibonacci heaps to run in time $O(m + n \log n)$.

   Now suppose there is a finite set C_m of m non-negative numbers, and the edge costs are given by a random permutation of the numbers in C_m on the edges in E. Show that under these conditions, the Jarnik-Prim algorithm can be implemented such that the expected number of decreaseKey operations is bounded by $O(n \log(m/n))$.

2. Suppose we want to maintain a minimum spanning tree of a graph under insertions and deletions of edges. For some undirected graph $G = (V, E)$ with edge costs, a minimum spanning tree T^* has been computed. Now a new edge e with cost $c(e)$ is added to G. Show how to compute the minimum spanning tree of the new graph from T^*. How much time does it take? How would you proceed if an edge was removed from G?

3. Let $G = (V, E)$ be a connected undirected graph with pairwise distinct edge costs. Design an algorithm that computes a spanning tree T of G whose maximal edge cost is as small as possible, i.e., that minimizes $\max_{e \in T} c(e)$. What running time can you achieve?