University of Konstanz
Department of Computer & Information Science
Sabine Cornelsen

Design and Analysis of Algorithms
Winter 2018/2019

Assignment 5

Post Date: 19 Nov 2018 **Due Date:** 26 Nov 2018, 11:30 am
You are permitted and encouraged to work in groups of two.

Problem 1: Minimum Spanning Tree — 7 Points

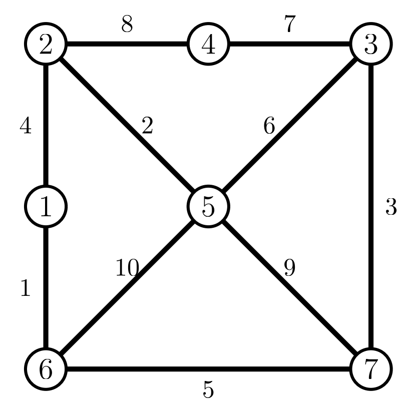
Find an MST for the graph on the right using

(a) the coloring method of Tarjan,

(b) the algorithm of Kruskal, and

(c) the algorithm of Prim.

Indicate in each step the edge that has to be colored together with the corresponding color. When you apply the algorithm of Prim, use a Fibonacci heap and give for each step the necessary heap operations and show how the heap looks like.



Problem 2: Height of a Fibonacci Heap — 4 Points

Prove or disprove that the height of a tree of a Fibonacci heap with n vertices is in $\mathcal{O}(\log n)$.

Problem 3: Cascading Cut — 9 Points

In the following we consider the Fibonacci heap from the lecture, but without cascading cut.

Let $T_k(i)$ be a Fibonacci heap with k root nodes with 0 to k children, where the root node with i children is missing. No tree has nodes of depth two or more. Further, the keys of the roots must decrease with increasing degree.

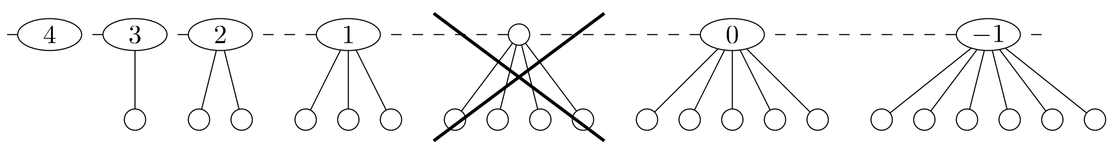


Figure 1: Example of a Fibonacci heap $T_6(4)$, where the root node with 4 children is missing.

(a) Given a heap $T_k(i), i = 1, \ldots, k$, show how to construct $T_k(i-1)$ using two Insert, one ExtractMin, and at most k DecreaseKey (without cascading cut) operations.

(b) Given a heap $T_k(0)$, show how to construct $T_{k+1}(k+1)$ with only one operation.

(c) Conclude that a heap $T_k(k)$ can be constructed with $\mathcal{O}(k^3)$ operations.

(d) Show that you can construct a $T_k(k)$ from a $T_k(k)$ by applying 1 Insert and 1 ExtractMin operations that need $\Omega(k)$ time.

(e) Conclude that the worstcase runtime is in $\Omega(m^{1+\epsilon})$ for m operations of Insert, ExtractMin, and DecreaseKey without cascading cuts for some $\epsilon > 0$.