Junior-Prof. Dr. Robert Elsässer, Marco Muñiz, Phillip Heidegger WS 2009/2010

Algorithms Theory, Assignment 4

http://lak.informatik.uni-freiburg.de/lak_teaching/ws09_10/algo0910.php

Submission: 17. Dec. 2009, 4 p.m.

- The solutions can be submitted in English or German
- You are required to submit your own solution.
- You are allowed to discuss your solutions with each other. Nevertheless, you are required to write down the answers in your own words.

Exercise 4.1 - Amortized Analysis [Points: 3+2]

1. For a data structure the i-th operation has costs c_i, where

$$c_i = \begin{cases} i & \text{if } i \in \{2^k \mid k \in \mathbb{N}\} \\ 1 & \text{else} \end{cases}$$

   Prove by the aggregate method that the worst cases amortized cost of an operation is constant.

2. Suppose we have a potential function Φ such that $\Phi_0 \neq 0$ and $\Phi_i \geq \Phi_0$ for all i. Show that there exists a potential function Φ' such that $\Phi'_0 = 0$, $\Phi'_i \geq 0$, the total amortized costs resulting from Φ' are the same as the total amortized costs resulting from Φ.

Exercise 4.2 - Ternary counter [Points: 3+2]

Consider a Ternary counter, i.e. a counter of base 3 with digits 0,1 and 2. The counter starts at 0 and will be increased by 1 n-times. The cost of increasing the counter from $i-1$ to i is given by the number of digits that must be changed. Let $A(n)$ denote the cost of incrementing the counter from 0 to n. Then:

1. Give the minimal $c \in \mathbb{R}$ such that

$$A(n) \leq cn \text{ for all } n \in \mathbb{N} \tag{1}$$

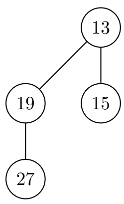
   holds. Use amortized analysis to show that c holds in equation 1.

2. Show that c is indeed minimal.
   Hint: Show that for every $\epsilon > 0$, $(c - \epsilon)$ in equation 1 does not hold.

Exercise 4.3 - Binomial queues [Points: 3+1]

Consider the binomial queue given below, and execute the following operations.

1. Q.insert(3), Q.insert(7), Q.insert(11), Q.insert(18), Q.insert(21), Q.insert(4), Q.decreaseKey(18,1) and Q.deleteMin().



2. By using the Child-Sibling-Representation, represent the binomial queue resulting from previous question 1

Exercise 4.4 - Trinomial Trees [Points: 6]

Define the family of trinomial trees analogously to the binomial trees. The family should have the following properties:

1. The number of nodes is 3^i for T_i.
2. The root of T_i has degree $2i$.
3. The hight of T_i is i.
4. There are $2^i \binom{n}{i}$ nodes with deeps i in T_n.

Describe formaly the structure of trinomial trees. Prove that the four properties above are fulfilled. Explain, how trinomial queues can be defined by using trinomial trees. What are the differences between binomial queues and trinomial queues? Describe the operation `Meld` for these queues.