Matthias Westermann, Marco Muñiz WS 2010/2011

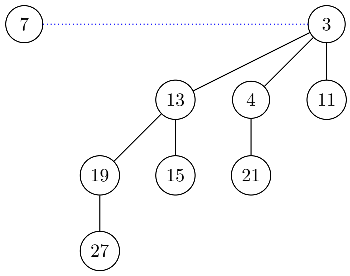
Algorithms Theory, Assignment 4

Submission: 15. Dec. 2010, 4 p.m.

Exercise 4.1 - Binomial queues [Points: 3+1]

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

1. Q.deleteMin(), Q.decreaseKey(27,1), Q.insert(8), Q.insert(9), Q.insert(10) and Q.deleteMin().



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

Exercise 4.2 - 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.

Exercise 4.3 - Fibonacci Heaps [Points: 5]

Execute the following operations on an initialy empty Fibonacci heap:

$$insert(12), insert(20), insert(30), insert(10), insert(25), insert(28),$$
$$insert(8), insert(26), insert(35), insert(40), insert(5), insert(50),$$
$$deleteMin(), decreasekey(26, 7), decreasekey(30, 6), deleteMin().$$

For all intermediate steps, illustrate the resulting Fibonacci heap. New elements should always be inserted to the left of the current minimum. The consolidation operation after $deleteMin()$ starts with the next element on the right hand side of the deleted minimum.

Exercise 4.4 - Fibonacci Heaps [Points: 5]

Show that the following claim is *not* true:

The maximum height of a tree within a Fibonacci Heap with n nodes is $\mathcal{O}(\log n)$.

Proceed as follows: For an arbitrary $n > 0$ give a sequence of operations that creates a Fibonacci heap finally consisting of one tree that is a linear chain of n nodes.