

## Assignment 2

**Post Date:** 31 Oct 2012    **Due Date:** 07 Nov 2012, 14:30

You are permitted and encouraged to work in groups of two.

### Problem 1: Binary Heap

**12 Points**

A *binary heap* is a complete binary tree, i.e., a binary tree in which every level, except possibly the lowermost, is completely filled, and, if the lowermost level is not complete, it is filled from left to right. Each node stores a value and fulfills the *heap property*, i.e., its value is smaller than or equal to the values of its children where the order of the two children is unspecified. Hence, the smallest value is always stored in the root.

To add a new node (INSERT), the node is first inserted at the leftmost free position of the lowermost level. After that, it is swapped with its current parent node as long as the current parent has a greater value, such that finally the heap property is recovered.

To delete the root node (DELETEMIN), the root is removed and the rightmost node of the lowermost level is moved to the root position instead. Similarly to the INSERT-operation, the new root is swapped downwards until the heap property is recovered. More precisely, it is swapped with one of its children, namely the one with the smallest value, as long as this value is smaller than its own.

- (a) Start with an empty heap and insert the following values or perform the operations in the given order: 6, 23, 14, 17, 13, 25, 2, DELETEMIN, 20, 21, 26, DELETEMIN  
Show the heap after each operation.
- (b) A heap is often realized as an array  $a[1..n]$  where  $a[1]$  is the root,  $a[2]$  the left child of the root,  $a[3]$  the right child of the root,  $a[4]$  the left child of  $a[2]$  etc.  
Formulate a condition for the array entries so that the heap property holds?
- (c) Show that the number of elements in a heap of height  $h$  is smaller than  $2^{h+1}$  but at least  $2^h$ . (Note that the height of a heap containing only the root is 0.)
- (d) Show that the height of a heap with  $n$  elements is  $\lfloor \log_2 n \rfloor$ .
- (e) Show that a heap with  $n$  elements contains at most  $\lceil \frac{n}{2^{h+1}} \rceil$  elements of height  $h$ . (Note that the height of the root is  $\lfloor \log_2 n \rfloor$ , and the height of the leaves is 0.)
- (f) Show that a heap with  $n$  elements can be build in  $\mathcal{O}(n)$  time.

[please turn over]

**Problem 2: Divide-and-Conquer Multiplication**

**8 Points**

- (a) Show how to multiply two linear polynomials,  $p(x) = p_0 + p_1x$  and  $q(x) = q_0 + q_1x$ , using only three multiplications.

**Hint:** One of the multiplications is  $(p_0 + p_1) \cdot (q_0 + q_1)$ .

- (b) Find a divide-and-conquer algorithm that multiplies two polynomials of degree  $n$  in  $\Theta(n^{\log_2 3})$ . You may assume that adding coefficients is a  $\mathcal{O}(1)$ -operation and that  $n$  is a power of 2.

**Hint:** Use part (a) to find an algorithm with running time  $T(n) = 3 \cdot T\left(\frac{n}{2}\right) + \mathcal{O}(n)$ .