CMPS 2200 Introduction to Algorithms – Fall 15

9/8/15

2. Homework

Due **9/17/15** at the beginning of the lab

1. **Min-Heaps (8 points)**
   Justify your answers shortly.

   (a) (2 points) Where is the maximum element located in a min-heap? How can you compute it, and what is the runtime?

   (b) (2 points) Is an array that is sorted in increasing order a min-heap? What about an array that is sorted in decreasing order?

   (c) (2 points) Can you give an example of a binary tree that is both a min-heap and a binary search tree? Find such an example for as many $n > 0$ as you can, where n is the number of elements in the binary tree. And argue for which n you cannot find such an example.

   (d) (2 points) Give a worst-case example of a min-heap that will cause **extract_min()** to run in $\Omega(\log n)$ time.

2. **d-Heaps (11 points)**
   A *d-ary min-heap*, d-heap for short, is the generalization of a binary heap to a d-ary tree, for a fixed $d \geq 2$. Every node can have up to d children, the tree has to be almost complete, and for every child of a parent the child's value is greater or equal than the parent's value.

   (a) Complete trees:

      i. (2 points) For given height $h = 2$, draw a complete 3-ary tree of height h and count the number n of nodes in it. Then draw a complete 4-ary tree of height h and count the number n of nodes.

      ii. (2 points) For given fixed height $h \geq 0$ and degree $d \geq 2$, give a formula for the number n of nodes in a complete d-ary tree of height h. Your formula should depend on n and d. (You do not need to prove your formula.)

   (b) (2 points) Suppose a d-heap is stored in an array (that begins with index 0). For an entry located at index i in which location is its parent and in which locations are its children?

   (c) (1 point) What is the height of a d-heap that contains n elements? The height should be a function of n and d. Shortly justify your answer (no formal proof needed).

FLIP OVER TO BACK PAGE ⟹

(d) (2 points) Shortly explain how the insertion procedure works for d-heaps (you do not have to give pseudocode). What is the runtime of inserting an element into a d-heap of n elements? The runtime should be a function of n and d.

(e) (2 points) Shortly explain how the extract_min procedure works for d-heaps (you do not have to give pseudocode). What is the runtime in terms of n and d, where n is the number of elements in the heap?

3. **Rotation (4 points)**

Assume that a tree node is an object that stores the data, a reference to the left tree node, and a reference to the right tree node. Give pseudo-code for a left-rotation. Use `parent` as an initial reference into the tree, and rotate at its left child.