Homework 3 of Computer Algorithms 2009

Dae-Ki Kang

May 16, 2009

Deadline: May 31, 2009

1. Show that there are at most $\lceil n/2^{h+1} \rceil$ nodes of height h in any n-element heap. The height of a node in a heap is the number of edges on the longest simple downward path from the node to a leaf.

   (Problem 6.3.3 (page 135) of CLRS, 2nd Ed.)

2. Problem 4.5 of DPV.
3. Problem 4.7 of DPV.
4. Problem 5.18 of DPV.
5. Problem 5.19 of DPV.
6. Problem 5.25 of DPV.
7. Problem 5.29 of DPV.
8. Problem 6.1 of DPV.
9. Problem 6.5 of DPV.
10. Problem 6.7 of DPV.
11. Consider a linked list data structure in C language as follows:

    ```
    struct NODE
    {
      int val;
      struct *NODE next;
    }
    ```

    You are given a linked list represented in the data structure above. Devise an algorithm that detects if the given linked list is cyclic or not. The algorithm should have $O(1)$ space complexity. (And analyze its running time in terms of asymptotic complexity with proof, if you can.)