Datastructures and Algorithms: Exam 1

October 22, 2013

You are allowed to use copies of the slides during the exam.

1. Consider the following definition of the power function:

   \[
   q(x, n) = \begin{cases} 
   1 & \text{if } n = 0 \\
   q(x, \frac{n}{2})^2 & \text{if } n > 0 \text{ is even} \\
   x \cdot q(x, \frac{n-1}{2})^2 & \text{if } n > 0 \text{ is odd}
   \end{cases}
   \]

   Give a pseudocode description of an algorithm \textit{Power}(x, n) to compute the power function according to this definition, for \( n \geq 0 \).

   Moreover, argue that the number of recursive calls of your program is in \( O(\log n) \).

   (20 pts)

2. Explain in detail how two stacks can be implemented using one array.

   Only if all places in the array are occupied, an overflow message may be given.

   (20 pts)
3. Consider below the Left-Right case of adding a node to an AVL tree. The root of the depicted subtree is the lowest unbalanced node, with balance factor $-2$, because the depth of subtree has $B$ increased, say from $h$ to $h + 1$.

![Diagram of an AVL tree](image)

Argue that subtrees $A$ and $C$ both have depth $h$. (20 pts)

4. Apply Kosaraju’s algorithm to find the strongly connected components in the following directed graph.

![Directed graph](image)

Consider two different scenario’s, where the first depth-first search starts in node $r$ or in node $u$. (20 pts)

5. Explain why it is crucial for the worst-case time complexity of Kruskal’s algorithm that in the disjoint-set data structure, when two disjoint sets are joined, the larger set subsumes the smaller set in the union-find data structure. What would be the worst-case time complexity of Kruskal’s algorithm if the smaller set subsumed the larger set? (20 pts)