Algorithms CMSC-27200/37000 Second Quiz. February 29, 2004

Instructor: László Babai

Name: ______________________________________ **U G** (circle one)

Show all your work. **Do not use book, notes, or scrap paper.** Write your answers in the space provided. This quiz contributes 5% to your course grade.

1. (5 points) Write a pseudocode for insertion in a binary search tree. Do NOT use recursive calls. Do not (re)balance the tree.

2. (2 points) Does the problem of sorting a list of integers belong to the complexity class P ? Reason your answer.

3. (6 points) Give a formal definition of the complexity class NP. You need to define what it means for a language $L \subseteq \Sigma^*$ to belong to NP. Use quantifier formalism, no English words (except for logical connectives such as "AND").

4. (3 points) For two sequences of real numbers, $\{a_n\}$ and $\{b_n\}$, define the relation $a_n \gtrsim b_n$ ("a_n is greater than or asymptotically equal to b_n").

5. (3 points) Prove: in order to arrange n data in a binary search tree, we need to perform $\gtrsim n \log n$ comparisons.

6. (3+3+3 points) For each statement, decide whether or not it is a loop-invariant for BFS: (a) "Vertex #2 is black." (b) "Vertex #2 is white." (c) "Vertex #2 cannot change from black to white." Reason your "NO" answers. Use the definition of loop-invariants given in a homework sheet.

7. (a) (3 points) What are the data maintained by a UNION-FIND data structure? (b) (3 points) What are the requests served and what is the effect of each request? (c) (5 points) Prove: under the hierarchical implementation with the "bigger wins" rule for merging, the depth of each tree remains $\leq \log n$. (d) (G only, 5 points) Prove: under the hierarchical implementation with the "deeper wins" rule for merging, the depth of each tree remains $\leq \log n$.

8. (G only) (a) (4 points) Describe the topology of Fibonacci heaps. (b) (4 points) Prove that the degree of each root node in a Fibonacci heap is $O(\log n)$. (c) (6 points) Describe the execution of the EXTRACT-MIN operation in a Fibonacci heap; state the actual cost and how it is covered.