

Algorithms CMSC-37000 Final Exam. March 14, 2006  
Instructor: László Babai

Show all your work. **Do not use text, notes, or scrap paper.** When describing an algorithm in pseudocode, **explain the meaning of your variables** (in English). This exam contributes 30% to your course grade. Take this problem sheet home.

1. (4 points) Given a digraph  $G$  by an array of adjacency lists, find an adjacency list representation of the reverse digraph in linear time. Describe your algorithm in pseudocode.
2. (12 points) (*Segmentation*) Consider the list of integers  $(1, 2, \dots, n)$ . A *segment* is a contiguous sublist,  $(i, i + 1, \dots, j)$  where  $i \leq j$ . A *segmentation* means splitting up the list into segments. For instance, for  $n = 8$ , a possible segmentation is  $(1, 2, 3), (4), (5, 6, 7, 8)$ . We are given a triangular array  $p(i, j)$  ( $1 \leq i \leq j \leq n$ ) of *segment penalties* (real numbers); the “penalty” of segment  $(i, \dots, j)$  is  $p(i, j)$ . The *cost* of the segmentation is the sum of the penalties of the segments. In the example, the cost is  $p(1, 3) + p(4, 4) + p(5, 8)$ . Find, in  $O(n^3)$  time, the optimal (min-cost) segmentation. Give your algorithm in pseudocode. Make sure you define your variables. Name the method used.
3. (8+4+5)
  - (a) Describe in pseudocode the algorithm that finds the median of  $n$  numbers using a linear number of comparisons.
  - (b) Let  $T(n)$  denote the number of comparisons used. State the recurrent inequality satisfied by  $T(n)$ .
  - (c) Prove that  $T(n) = O(n)$ .
4. (6+2+8+8 points)
  - (a) Prove that the depth of a sorting network (number of parallel steps) is  $\gtrsim 2 \log n$  (where  $n$  is the number of items to be sorted).
  - (b) Name the authors of a sorting network of depth  $O(\log n)$ .
  - (c) Describe in pseudocode Batcher’s “odd-even” MERGE network. (Do NOT describe the resulting sorting network.)

- (d) Prove the correctness of the algorithm. (You may use the (0,1)-law without proof.)
5. (5 points) Given a digraph  $G$  by an array of adjacency lists, decide in linear time whether or not  $G$  is strongly connected.
6. (5 points) Write a 3-CNF formula that is not satisfiable. Within each clause, the three literals must correspond to three distinct variables.
7. (3+6 points)
- (a) Name a significant computational task discussed in class, other than mergesort, which led to the recurrent inequality  $T(n) \leq 2T(n/2) + O(n)$ .
- (b) State and prove the asymptotic solution of this recurrence.
8. (a) (6 points) Define the concept of a “loop invariant.” Be as formal as reasonable. Make sure you give a clear definition of what kind of statement can be a candidate loop invariant. Include the definition of the domain and range of the predicates and transformations (functions) involved. Don’t forget to define the loop to which you are referring.
- (b) (3+3+3 points) Decide which of the following statements are loop-invariants for Dijkstra’s algorithm. Reason your answers. (b1) All black vertices are accessible. (b2) All accessible vertices are black. (b3) All accessible vertices will eventually become black.
9. (6+6+10B points)
- (a) Define the concept of a universal family of hash functions.
- (b) Give infinitely many examples of universal families of hash functions.
- (c) (BONUS) Describe a randomized algorithm to find the nearest pair among  $n$  points in the plane in expected  $O(n)$  steps.
10. (6+6 points)
- (a) Give a formal definition of the complexity class NP. You may assume that the definition of the complexity class P is known. Your definition begins with the phrase  
*“The language  $L \subseteq \Sigma^*$  belongs to NP if and only if”*  
 and you have to continue with an expression which contains no English words except possibly logical connectives such as “if,” “then,” “and.” Almost all credit is lost if your formula defines the wrong language class due to a misplaced or omitted quantifier.

- (b) An incorrect solution to part (a) is this:  
 $(\exists \Sigma_1)(\exists L_1 \subseteq \Sigma_1^*, L \in P)(\forall x \in \Sigma^*)(x \in L \Leftrightarrow (\exists c)(\exists w \in \Sigma_1^*)(|w| \leq |x|^c \text{ and } (x, w) \in L_1))$ .

Name the complexity class defined by this formula. Give a brief indication why.

11. (3+7+5+4+5)

- (a) Define the notion of Karp-reduction. Don't forget to state what is being reduced to what.
- (b) Describe a Karp-reduction from 3-SAT to CLIQUE.
- (c) Describe a Karp-reduction from 3-COL to 4-COL, where  $k$ -COL denotes the set of  $k$ -colorable graphs.
- (d) Describe a Karp-reduction from 3-COL to HALTING.
- (e) Prove: there is no Karp-reduction from HALTING to 3-COL.

12. (8+3+12B)

- (a) Recall that the topology of an AVL tree is a binary tree (each node has  $\leq 2$  children such that the heights of the left and right subtrees at each node differ by at most 1. Prove that the height of an AVL tree with  $n$  nodes is  $\lesssim \log n / \log \phi$  where  $\phi = (1 + \sqrt{5})/2$  is the golden ratio.
- (b) The AVL tree is a binary search tree. In what order do we store the data at the nodes of a binary search tree?
- (c) (BONUS) Given a list of data  $a_1, \dots, a_n$ , let RANGESUM( $r, s$ ) denote the quantity the sum of those  $a_i$  which satisfy the inequalities  $r \leq a_i < s$ . Describe what additional data need to be maintained at the nodes of an AVL tree so as to permit serving RANGESUM queries in  $O(\log n)$  steps. Describe how to update these additional data under INSERT (no rotation).