SPRING 1993
COMPUTER SCIENCES DEPARTMENT
UNIVERSITY OF WISCONSIN—MADISON
PH. D. QUALIFYING EXAMINATION

Foundations of Computing
Breadth Examination

Monday, February 8, 1993
3:00 – 5:00 PM
113 Psychology

GENERAL INSTRUCTIONS:

1. Answer each question in a separate book.
2. Indicate on the cover of *each* book the area of the exam, your code number, and the question answered in that book. On *one* of your books list the numbers of *all* the questions answered. *Do not write your name on any answer book.*
3. Return all answer books in the folder provided. Additional answer books are available if needed.

SPECIFIC INSTRUCTIONS:

Answer any *three* of the following *five* questions.

POLICY ON MISPRINTS AND AMBIGUITIES:

The Exam Committee tires to proofread the exam as carefully as possible. Nevertheless, the exam sometimes contains misprints and ambiguities. If you are convinced a problem has been stated incorrectly, mention this to the proctor. If necessary, the proctor can contact a representative of the area to resolve problems during the *first hour* of the exam. In any case, you should indicate your interpretation of the problem in your written answer. Your interpretation should be such that the problem is nontrivial.

Theory of Computing
Breadth Exam
Spring 1993

Directions: There are five questions, of which you must do THREE.

1. Let M be a deterministic finite automaton. Suppose that x is an *infinite* word made up of letters from Σ, the input alphabet of M. More precisely, x is an infinite sequence of elements chosen from Σ. We say that M *accepts* x if, when started in its initial state and given the letters of x in order, M passes through an accepting state infinitely many times. The language accepted by M is the set of all infinite words which M accepts. Finite automata used in this fashion are called *ω-automata* in the literature.

   State and prove a generalization of the "pumping lemma" for ω-automata; i.e. applicable to the above situation.

2. The Euclidean Traveling Salesman problem is the problem of finding a shortest tour of n distinct points on the plane. (For simplicity, assume no two points have the same x coordinate). It is well known that the Euclidean Traveling Salesman problem is NP-hard.

   Suppose we restrict our attention to "bitonic" tours, that is, tours which start at the leftmost point, go strictly left to right to the rightmost point, and then go strictly right to left back to the starting point.

   Describe an $O(n^2)$ time algorithm for determining an optimal bitonic tour. [Hint: dynamic programming.]

3. Consider a parallel computer with a large number of processors. In one time step, a processor can do exactly one of the following operations: send a message to some other processor, or add two numbers. Your objective is to form the sum of as many numbers as possible in m time steps. At the end of your algorithm, one processor should know the result (the sum). You are allowed to use as many processors as you like, and you may assume any suitable distribution of the inputs among processors; inputs are not duplicated.
   a) How many numbers can you add in m steps using a "binary tree" style algorithm?
   b) Can you do better?

4. Given a set Q of points in the plane, we define the *convex layers* of Q inductively. The first convex layer of Q consists of those points in Q that are vertices of the convex hull of Q, denoted by $\mathrm{CH}(Q)$. For $i > 1$, define Q_i to consist of those points in Q with all points in convex layers $1, 2 \ldots, i-1$ removed. Then, the ith convex layer of Q is $\mathrm{CH}(Q_i)$ if Q_i is not empty, and is undefined otherwise.
   a) Give an $O(n^2)$ time algorithm to find the convex layers of a set of n points. Your algorithm should output the points in each layer in their natural order, say clockwise.
   b) Give an $\Omega(n \log n)$ lower bound for this problem.

5. A context free grammar is in *Chomsky normal form* if all productions are of the form $X \to YZ$ or $X \to a$ where X, Y, Z are non-terminals and a is a terminal. Define the *size* of a grammar to be the number of nonterminals plus the number of terminals.

   a) Convert the following grammar into Chomsky Normal Form:

$$S \to CB$$

$$S \to CBBC$$

$$C \to aaa$$

$$B \to b.$$

   b) Give a method for converting grammars to Chomsky normal form. For your method, how large can the converted grammar be, if the original grammar has size n? Give an example showing your estimate is best possible up to constant factors.