Fall 2001
Qualifying Exam
Foundations of Computing

Directions. There are five questions. Answer any three of them.

1. We say that a square matrix is *unimodular* if its determinant is 1. Show the following problem is NP-hard:

> **Instance:** A set of generators $G_1, G_2, \ldots, G_m$, and a target matrix T, all unimodular 2 by 2 matrices with integer entries. (All entries are written in binary notation.)
>
> **Problem:** Is T a product of a sequence of zero or more G_i's, where repetition is allowed?

You may assume the following subset sum problem is NP-complete:

> **Instance:** A set of positive integers, and target t, all given in binary.
>
> **Problem:** Is t a sum of a subset of these binary integers?

2. For this problem you are to consider the Boolean function PARITY, which takes n 0-1 inputs and returns their sum mod 2. Using AND, OR, MAJORITY (of unbounded fan-in) and NOT gates, implement a polynomial size, constant depth, Boolean circuit for PARITY. Hint: implement threshold gates first.

3. Define the *determinant* and the *permanent* of a square matrix M over a commutative ring R. Your definition should exhibit the close similarity between these two concepts.

For each of the following computational problems, give the minimal complexity classes in which you can put them. Briefly explain. The input is always an integer matrix M with all entries in $\{0, 1\}$.

a. Determining whether the permanent of M is zero or not.
b. Computing the permanent of M.
c. Determining whether the permanent of M is odd or even.

4. Professor Lavoisier has been tasked with preparing the punch for this fall's CS party. Midway through assembly, he realizes that the ingredients will not fit into the bowl provided, so he gets another one for the remaining ingredients. He now has two bowls with the same amount of material but differing in composition. There is now the task of making the two bowls taste the same, which he can do by ladling a fraction $\alpha \ll 1$ of each of the mixtures back and forth.

The goal is to make the amounts of each ingredient in the two bowls differ by less than ϵ. Show that it suffices to do

$$O\left(\frac{\log(\epsilon^{-1})}{\alpha}\right)$$

operations. (An operation consists of ladling from the first bowl to the second, then from the second back to the first.)

5. Let G be a digraph with n vertices. Recall that a *Hamiltonian cycle* is a sequence of distinct vertices $v_0, v_1, \ldots, v_n$ such that (v_i, v_{i+1}) is an edge for $0 \leq i < n$, and $v_0 = v_n$. Present a dynamic programming algorithm with complexity $o(n!)$ to determine if G has a Hamiltonian cycle.