Theoretical Computer Science
Qualifying Exam
Spring 1994

Directions. This exam contains seven problems. Please do any four.

Problem 1. Let X_i, $i = 1, \ldots, n$, be independent random variables. Assume that $X_i = 0$ with positive probability p_i and $X_i = \alpha_i > 0$ with positive probability $q_i = 1 - p_i$. For this problem you are to consider algorithms for computing the distribution of $Y = \sum_{i=1}^{n} X_i$.

a) Suppose you are given a list of n pairs (α_i, p_i). Your goal is to produce another list of pairs (β_i, r_i) such that: a) $Y = \beta_i$ with probability r_i, and b) the list is sorted, so that $\beta_1 \leq \beta_2 \leq \beta_3 \leq \cdots$. Give an algorithm for this purpose taking $O(2^n)$ steps. (Note that this is best possible, because there are inputs for which Y's distribution will have 2^n mass points.)

b) Under the same requirements as a), find an algorithm that takes time polynomial in the actual length of the output. Try to get as efficient an algorithm as you can.

The next two problems will rely on the following definitions.

Let N be a polynomial time bounded nondeterministic Turing machine. We will say that N is precise if two things are true about its behavior. First, there is a polynomial p such that for any input of length n, N runs for exactly $p(n)$ steps before halting. Secondly, at each step N has exactly 2 nondeterministic choices.

A language L is in the class RP if there is a precise Turing machine N such that,

1) If $x \in L$, then at least $1/2$ of the $2^{p(|x|)}$ computations of N on input x halt and say "yes;"
2) If $x \notin L$, then all computations of N on input x halt and say "no."

A language L is in the class BPP if there is a precise Turing machine N such that,

1) If $x \in L$, then at least $3/4$ of the $2^{p(|x|)}$ computations of N on input x halt and say "yes;"
2) If $x \notin L$, then at least $3/4$ of the $2^{p(|x|)}$ computations of N on input x halt and say "no."

A language L is in the class PP if there is a precise Turing machine N such that $x \in L$ if and only if more than half of N's computations on input x halt and say "yes."

Problem 2. Consider the complexity classes RP, BPP, and PP, and the set operations of Union, Intersection and Complement. For each class/operation pair, try to prove or disprove that the class is closed under the operation. For any case in which you are unable to give a proof, explain why it seems difficult to settle and conjecture what the answer is. [Note: you must provide at least one rigorous proof to get any credit for this problem.]

Problem 3. Show that every language that is in RP can be decided by a collection of polynomial size circuits.

Problem 4. According to reputable sources (e.g. Usenet, The New York Times), Andrew Wiles has recently shown that there are no positive integers x, y, z, n with $n \geq 3$ for which

$$x^n + y^n = z^n.$$

You receive a letter from a rich but skeptical individual, containing four huge numbers x, y, z, n that are asserted to satisfy the above equation. The letter offers a large reward to anyone who can disprove this assertion.

a) Explain why it is infeasible to do this by simply plugging the numbers into the formula.

b) Give a polynomial time algorithm that will allow you to collect the reward, assuming that Wiles is correct and the offer is genuine.

Problem 5.

a) Pick a non-regular language, and describe a 1-tape deterministic Turing machine that recognizes your language in $O(n \log n)$ time. (In this model, the input tape is a read/write tape with a single head, and is the only tape the machine has.)

b) Pick a non-regular language, and describe a 1-tape nondeterministic Turing machine that recognizes this language in $o(n \log n)$ time.

Problem 6. The graph N-coloring problem, a variation of the classical graph coloring problem, is defined as follows. The input is a graph G and an integer $N \geq 2$; the problem is to color the nodes of G with at most N colors, so that the number of conflicting edges are minimized. A conflicting edge is one with both endpoints colored by the same color.

a) Show that this is an NP-hard problem.

b) Consider the following local search algorithm for the problem.

> Initially, color the nodes arbitrarily.
>
> Repeat the following step until it is no longer possible: Find a node v and a color c such that by changing the color of v to c, the number of conflicting edges can be reduced. Color v with color c.
>
> Output the resulting coloring.

Show that this algorithm can be implemented in polynomial time, and that on all inputs, the output will have at most $|E|/N$ conflicting edges.

Problem 7. Let X and Y be sets of size n. Show how to sort the set $X + Y = \{x + y \mid x \in X, y \in Y\}$, using $O(n^2)$ comparisons. [Hint: Use the equivalence $(x_i + y_j \leq x_k + y_l) \equiv (x_i - x_k \leq y_l - y_j)$.]