Algorithms – CS-27200/37000 Homework – February 20, 2004
Instructor: László Babai Ry-164 e-mail: laci@cs.uchicago.edu

ADVICE. Take advantage of the TAs' office hours Monday, Tuesday and Thursday 5–6pm in the Theory lounge (Ry–162).

DATES TO REMEMBER. Mon Feb 23: Quiz 2; Mon Mar 8: Midterm 2.

15.1 (0 points – do not hand in)

(a) Let A and B be $k \times k$ matrices with m-digit integers as entries. Show that the matrix $C = AB$ can be computed in polynomial time. Compute the smallest exponent without invoking sophisticated multiplication of numbers or matrices.

(b) The problem stated in (a) does *not* belong to the complexity class P. Why?

15.2 A linear inequality in the variables x, y, z is an inequality of the form $3x - 5y + z \leq 6$. (We may replace the coefficients $3, -5, 1, 6$ by arbitrary real numbers, and the number of variables may also be arbitrary. We shall assume, however, that *all coefficients are integers.*) The *linear programming* (LP) problem takes a system of linear inequalities as input (m inequalities in n variables) and asks its *feasibility* (does there exist a solution, satisfying all the given inequalities at the same time?). The *integer linear programming* (ILP) problem asks the existence of a solution in *integers* (each variable must take an integral value). A $(0, 1)$-ILP asks the existence of a solution where each variable takes the value 0 or 1.

LP is one of the most widely applied algorithmic problems. **Fact.** LP is solvable in polynomial time (Khachiyan, 1979, Karmarkar, 1983).

(i) (2 points) Give an example of an LP (with integer coefficients) which is feasible (has solution(s)) but which is not feasible as an ILP. Use as few variables as possible.

(ii) (4 points) Same as item (i) but all coefficients in the LP must be 0, 1, or -1, including the numbers on the right hand side. (A correct solution to this question also earns you the 2 points for part (i) so no separate solution to (i) is required to get all the 6 points.)

(iii) (5 points) Prove that (0,1)-ILP belongs to NP. State the (polynomial-time verifiable) witness (of the yes-answers). Indicate why it is verifiable in polynomial time (length of input to verification algorithm, estimated running time of verification).

(iv) (G only, 6 points) Is it evident that ILP belongs to NP? Argue your answer. Be as specific about the potentially difficult technical detail as you can.

(v) (10 points, due Wednesday, February 25) Prove that (0,1)-ILP is at least as hard as 3-COL (graph 3-colorability) by constructing a Karp-reduction from 3-COL to (0,1)-ILP. (Note: once we know that 3-COL is NP-complete, this reduction will prove that (0,1)-ILP is NP-complete.) *Comment.* Recall from class the concept of Karp-reduction: Given an instance of 3-COL (i. e., a graph G), you need to construct in polynomial time an instance of (0,1)-ILP (a system of linear inequalities as above) which is feasible if and only if G is 3-colorable.

15.3 (8 points) Let k-COL denote set of all k-colorable (undirected) graphs. Under suitable encoding of graphs as strings over a finite alphabet, k-COL is a language. Assuming that 3-COL in NP-complete (which it is), prove that 4-COL is NP-complete. State clearly, which language you are reducing to which language. *Hint.* The reduction will be very simple, only a few lines to describe. Make sure after describing the reduction you prove all that needs to be proved. Before proving something, state clearly what exactly you are proving. Conceptual clarity is paramount.

15.4 (6 points, due Wednesday, February 25) Let L_1 and L_2 be two languages, $L_i \subseteq \Sigma_i^*$. Assume $L_1 \prec L_2$ (L_1 is Karp-reducible to L_2); the reduction function $f : \Sigma_1^* \to \Sigma_2^*$ runs in time $O(n^c)$. (Time = cost = number of bit-operations.) Prove: if L_2 is recognizable in time $O(n^{\log n})$ then L_1 is recognizable in time $O(n^{O(\log n)})$. Explain why a big-Oh appeared to the exponent; state the smallest constant hidden in the big-Oh notation in the exponent.