CSE 531 Final EXAM

December 11, 1995

1. (25%) Let L_1 and L_2 be accepted by NFA's $M_1 = (Q_1, \Sigma_1, \delta_1, q_{10}, F_1)$ and $M_2 = (Q_2, \Sigma_2, \delta_2, q_{20}, F_2)$ respectively. Assume $\Sigma_2 \subseteq \Sigma_1$. For $x \in \Sigma_1$, define $h(x)$ to be the string in Σ_2 where each symbol of $\Sigma_1 - \Sigma_2$ is erased in x. For example, if $\Sigma_1 = \{a, b, c\}$ and $\Sigma_2 = \{a, b\}$, then h(abccacb) = abab. Define

$$L_1 \mid L_2 = \{x : x \in L_1 \text{ and } h(x) \in L_2\}.$$

   (a) For $L_1 = ((a+b)^*c)^*$ and $(bc)^*$ define $L_1 \mid L_2$ as a regular expression.

   (b) Use a cross product construction to define a finite automaton M of some variety which accepts $L_1 \mid L_2$.

   (c) State a behavioral lemma for your construction which could be used to show $L(M) = L_1 \mid L_2$.

2. (25%) Consider a k-pebble finite automaton. Such a finite automaton is deterministic, has a single read-only tape, and a single read head which can move in two directions. However, the machine has k distinguishible pebbles which it can leave a pick up. If the machine is holding a pebble then it can place it down on the current cell it is visiting. If the head is over a cell which contains a pebble then it can pick it up. There are end markers at both ends of the input to prevent the machine from running off the tape.

   (a) Describe informally a two-pebble finite automaton which accepts the language $\{a^n b^n : n \geq 0\}$.

   (b) Why is the language accepted by a zero-pebble automaton regular?

   (c) Demonstrate why the emptiness problem for two-pebble automata is undecidable, by showing how such a machine can accept strings which represent accepting computations of a Turing machine.

   (d) Argue that if L is accepted by a k-pebble automaton then $L \in \text{DSPACE}(\log n)$.

3. (25%) Zero-one integer programming is the problem of determining if a system of linear inequalities has a solution where each of the variables is restricted to be either 0 or 1. For example:

$$\begin{aligned} 3 &\geq 2x + y \\ 1 &\leq x - 5y + z \\ 7 &\geq 2y + 4z \end{aligned}$$

is such a set which has a solution $x = 1, y = 0, z = 1$.

   (a) Show that 3-CNF Satisfiablity is reducible in polynomial time to zero-one integer programming.

   (b) Demonstrate that zero-one integer programming is NP-complete.

4. (25%) An alternating Turing machine $M = (U, E, \Sigma, \Gamma, \delta, q_0, B, F)$ is such that there are two kinds of states, universal (U) and existential (E). For a configuration C, we define C *leads to acceptance* recursively as follows: either (i) C is a configuration whose state is in F or (ii) C is a configuration whose state is universal and for all D such that $C \vdash_M D$, D leads to acceptance, or (iii) C is a configuration whose state is existential and for some D such that $C \vdash_M D$, D leads to acceptance. Finally, define

$$L(M) = \{x \in \Sigma^* : q_0 x \text{ leads to acceptance}\}.$$

   (a) Argue that $\text{ASPACE}(s(n)) \subseteq \bigcup_c \text{DTIME}(c^{s(n)})$ whenever $s(n) \geq a \log n$ for some $a > 0$. Hint: how many distinct configurations are there of an alternating Turing machine which uses no more than $s(n)$ storage?

   (b) Why is $\text{ASPACE}(\log n) \subseteq P$?

   (c) It is also the case that $\bigcup_c \text{DTIME}(c^{s(n)}) \subseteq \text{ASPACE}(s(n))$ whenever $s(n) \geq a \log n$ for some $a > 0$. From this fact why might it be conceivable, and perhaps likely, that the problem of determining if a player has a winning strategy from a board position in an $n \times n$ board game requires exponential time? Think of the board game being as a two-player game such as go, checkers, or chess generalized in some way.