CSE 531 FINAL EXAM

December 10, 1993

1. (15%) Define $K = \{x : x \in \{0,1\}^*, x \in L(T_x)\}$. Prove by contradiction that K is not computable.

2. (15%)

   (a) Carefully define what it means for $A \leq_m B$.

   (b) Argue that if $A \leq_m B$ then $\bar{A} \leq_m \bar{B}$.

   (c) Argue that if $A \leq_m B$ and B is recursively enumerable, then so is A.

   (d) Show that the language $L = \{x : L(T_x) \text{ is infinite}\}$ is not recursively enumerable.

3. (15%) Consider the following problems:

   (a) Input: A PDA M
   Property: M is deterministic

   (b) Input: A PDA M
   Property: $L(M)$ is a deterministic context-free language

   (c) Input: A context-free grammar G
   Property: G is ambiguous

   (d) Input: A context-free grammar G
   Property: $L(G)$ is inherently ambiguous

   Which of the problems are decidable and which are not.

   Define M_G to be the PDA produced by the standard top down construction of a PDA from a context-free grammar G. Define G_M to be the grammar produced by the standard construction of grammar from a PDA M.

   For each of the following statements relating the above problems determine if they are true or false. For each case that is false give a counter example.

   (a) $L(M)$ is deterministic context-free implies M is deterministic.

   (b) M is deterministic implies the G_M is not ambiguous.

   (c) G ambiguous implies $L(G)$ is inherently ambiguous.

   (d) $L(G)$ is inherently ambiguous implies G is ambiguous.

   (e) G is not ambiguous implies that M_G is deterministic.

4. (10%) Consider the "not everything" problem for regular expressions without star. We call this problem NE$(+,\cdot)$.

   Input: A regular expression α over $\{0,1\}$ which just uses union and concatenation (no Kleene star) and an integer k.

   Property: $L(\alpha) \neq \{0,1\}^k$

   Argue carefully that NE$(+,\cdot)$ is in NP.

5. (15%) Define a two-headed finite automaton (2-DFA) to be a finite automaton with two read heads. A move of the automaton consists of reading the two symbols under the read heads, changing state, and moving one or both of the heads to the right. The machine accepts an input if it ever enters an accepting state.

   (a) Explain briefly how a 2-DFA can accept the language $L = \{xcxd : x \in \{0,1\}^*\}$.

   (b) Argue that it is undecidable if $L(M) = \phi$, given a 2-DFA M.

6. (15%) Let $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$ be a deterministic Turing machine.

   (a) Define a regular expression of length $O(m^2)$ which defines the language $L = \{ucvd : |u| = |v| = m \text{ and } u \vdash_M v\}$. Assume that the symbols c, d are not in the set $Q \cup \Gamma$.

   (b) Suppose M runs in space cn^k for some c and k. Given an input x of length n, describe two regular expressions α and β each of length polynomial in n, such that $x \in L(M)$ if and only if $L(\alpha) \cap L(\beta) \neq \phi$.

   (c) Why is the "non-empty intersection problem" for regular expressions PSPACE-complete? Recall that a language A is PSPACE-complete if $A \in$ PSPACE $= \bigcup_{c,k}$DSPACE(cn^k) and for all $B \in$ PSPACE, $B \leq_m^P A$.

7. (15%) If A and B are two languages define join$(A,B) = \{z : z = uvx$ for some u,v, and x, and $uv \in A$ and $vx \in B\}$. If M_A and M_B are two DFA's accepting A and B respectively, give the construction of some kind of finite automaton M which accepts join(A,B). Explain what your automaton does. Hint: some combination of a construction used for intersection and a construction used for concatenation will do the trick.