PhD Qualifying Exam - Foundations

August 22, 2006

1. Let Σ be a finite alphabet. Prove that Σ^* is a countable set.

2. Let δ be the transition function of a deterministic finite-state automaton $M = (Q, \Sigma, \delta, q_0, F)$:

   $$\delta : Q \times \Sigma \to Q.$$

   Consider the extension

   $$\hat{\delta} : Q \times \Sigma^* \to Q,$$

   of δ, defined by

   $$\begin{aligned} \hat{\delta}(q, \epsilon) &= q \\ \hat{\delta}(q, aw) &= \hat{\delta}(\delta(q, a), w) \qquad \text{for } a \in \Sigma. \end{aligned}$$

   (a) Prove that $\hat{\delta}(q, a) = \delta(q, a)$ for all $q \in Q, a \in \Sigma$.

   (b) Prove by induction on strings that for each pair of strings $x, y \in \Sigma^*$ and state $q \in Q$, $\hat{\delta}(q, xy) = \hat{\delta}(\hat{\delta}(q, x), y)$.

3. Let Σ be an alphabet. For $a \in \Sigma$, let $\#_a : \Sigma^* \to \mathbf{N}$ be the function that gives the number of times the character a occurs in the input string.

   (a) Prove that $L = \{w \in \{a, b\}^* \mid \#_a(w) = \#_b(w)\}$ is not regular.

   (b) Show that L is context-free. (You need not prove that your construction yields L.)

4. (a) Show that the class of context-free languages is closed under set union.

   (b) Show that the class of context-free languages is not closed under set intersection.

5. Let $L_{CFL} = \{\langle M \rangle \mid \ M$ a Turing machine that accepts a context-free language $\}$.

   Prove that L_{CFL} is undecidable. Your proof can not make use of Rice's Theorem. [Hint: use a reduction argument.]

6. Let

   $A_{DFA} = \{\langle M, w\rangle \mid M \text{ a deterministic finite-state automaton that accepts string } w\ \}$

   (a) Outline a proof that A_{DFA} is decidable in polynomial time.

   (b) Prove that A_{DFA} is not accepted by a finite-state automaton.

7. Let a "funny graph" be a finite directed-graph where each vertex is either "square" or "circular" (but not both), and each edge is between a square vertex and a circular vertex (in either order). Prove that the following problem is NP-complete.

   - Given a funny graph, is there a way to color the circular vertices red or green, so that every square vertex either has an out-edge to a green vertex, or an in-edge from a red vertex.