CS5371 Theory of Computation

Homework 5
Due: 3:20 pm, January 5, 2007 (before class)

1. Let S be a set and let C be a collection of subsets of S. A set S', with $S' \subseteq S$, is a called a *hitting set* for C if every subset in C contains at least an element in S'. Let *HITSET* be the language
$$\{\langle C, k\rangle \mid C \text{ has a hitting set of size } k\}.$$
Prove that *HITSET* is NP-complete.

2. Let U be the language
$$\{\langle M, x, \#^t\rangle \mid \text{TM } M \text{ accepts input } x \text{ within } t \text{ steps on at least one branch}\}.$$
Show that U is NP-complete. (For this problem, you are required to prove it without using reduction from any known NP-complete problems.)

3. We say a language A is in coNP if its complement, $\overline{A}$, is in NP. We call a regular expression *star-free* if it does not contain any star operations. Let EQ_{SF_RFX} be the language
$$\{\langle R, S\rangle \mid R, S \text{ are equivalent star-free regular expressions}\}.$$
Show that EQ_{SF_RFX} is in coNP. Why does your argument fail for general regular expressions?

4. (Choose either Q4 or Q5.) Show that the following problem is NP-complete. You are given a set of states $Q = \{q_0, q_1, \ldots, q_\ell\}$ and a collection of pairs $\{(s_1, r_1), \ldots, (s_k, r_k)\}$ where the s_i are distinct strings over $\Sigma = \{\mathtt{0}, \mathtt{1}\}$, and the r_i are (not necessarily distinct) members of Q. Determine whether a DFA $M = (Q, \Sigma, \delta, q_0, F)$ exists where $\delta(q_0, s_i) = r_i$ for each i. Here, the notation $\delta(q, s)$ stands for the state that M enters after reading s, starting at state q. (Note that F is irrelevant here).

5. (Choose either Q4 or Q5.) Consider the algorithm *MINIMIZE*, which takes a DFA M as input and outputs DFA M'.

   MINIMIZE = "On input $\langle M\rangle$, where $M = (Q, \Sigma, \delta, q_0, A)$ is a DFA:

   1. Remove all states of M that are unreachable from the start state.
   2. Construct the following undirected graph G whose nodes are the states of M.
   3. Place an edge in G connecting every accept state with every nonaccept state. Add additional edges as follows.
   4. Repeat until no new edges are added to G:
   5. For every pair of distinct states q and r of M and every $a \in \Sigma$:
   6. Add the edge (q, r) to G if $(\delta(q, a), \delta(r, a))$ is an edge of G.
   7. For each state q, let $[q]$ be the collection of states
   $$[q] = \{r \in Q \mid \text{ no edge joins } q \text{ and } r \text{ in } G\}.$$

8. Form a new DFA $M' = (Q', \Sigma, \delta', q_0', A')$ where
   - $Q' = \{[q] | q \in Q\}$, (if $[q] = [r]$, only one of them is in Q'),
   - $\delta'([q], a) = [\delta(q, a)]$, for every $q \in Q$ and $a \in \Sigma$,
   - $q_0' = [q_0]$, and
   - $A' = \{[q] | q \in A\}$.
9. Output $\langle M' \rangle$."

A. Show that M and M' are equivalent.

B. Show that M' is minimal—that is, no DFA with fewer states recognizes the same language. You may use the Myhill-Nerode Theorem.

C. Show that *MINIMIZE* operates in polynomial time.