CSE 322 Spring 2005
Assignment #8

Due: Friday, June 3, 2005

Reading assignment: Read Chapter 4 of Sipser's text.

Problems:

1. Sipser's text, page 169, Exercise 4.7. (Note that infinite binary sequences are not strings since any string has finite length.)

2. Sipser's text, page 169, Exercise 4.8.

3. Sipser's text, page 169, Exercise 4.5.

4. Sipser's text, page 169, Problem 4.11.

5. Define a *queue* automaton $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$ where Q is the finite set of *states*, Σ is the *input alphabet*, Γ is the *queue alphabet*, q_0 is the *start state*, q_{accept} and q_{reject} are *accept* and *reject* states respectively, and

   $$\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times (\Gamma \cup \{\varepsilon\}) \to Q \times (\Gamma \cup \varepsilon).$$

   A *configuration* of a queue automaton is an element of $Q \times \Sigma^* \times \Gamma^*$; configuration (q, y, z) represents that the current state is q, the remaining input is y, the current contents of the queue is z (with the left-most character on the left end of z).
   If $\delta(p, a, A) = (q, B)$ where $A, B \in \Gamma \cup \{\varepsilon\}$ then its action on configurations is to take (p, ay, Az) to (q, y, zB).
   That is, a queue automaton is like a DPDA except that it has a queue instead of a stack.

   Sketch how queue automata are equivalent to Turing machines.
   (HINT: to simulate one step of the TM might require going through the entire queue of the queue automaton.)

6. (Bonus) Sipser's text page 170, Problem 4.20

7. (Bonus) Sipser's text page 169, Problem 4.9