CSCI2243-Assignment 9

Assigned December 3 due Saturday, December 9

In connection with Problem 4, I will supply the code for the Turing machine simulator that I demonstrated in class. You are not obliged to use this, but you might want to play with it a little, and it could help in answering the question.

1. Do parts (a)-(e) of Exercises 1 and 2 of of Chapter 8. For these relatively elementary problems, you do not need to deploy the algorithms for converting regular expressions into automata and vice-versa. You just kind of figure it out.

2. This is a simplified version of Exercise 9 of Chapter 8. You don't have to prove anything, just construct some automata.

(a) Consider the language represented by the regular expression $(a+b)^2 b(a+b)^*$. This is the set of all words in $\{a,b\}^*$ whose third letter is b. Draw the state-transition diagram of a DFA that recognizes this language. The smallest such DFA has 5 states, but one of them is a 'dead state' that can simply be eliminated, producing an equivalent NFA.

(b) Now draw the state diagram of a 4-state NFA that recognizes the language corresponding to the regular expression $(a+b)^* b(a+b)^2$. This is the set of all strings in which the third letter counting from the *right* is b. (If you use the constructions outlined in the text, you will get a larger NFA, because these algorithms require the insertion of ϵ-transitions. But in this particular example there is no need to put in such transitions; you can just glue the pieces together directly.)

(c) Finally, use the subset construction to build a DFA recognizing this language. Show the tabulation of the states of the DFA carefully, and draw the state-transition diagram as neatly as you can.

4. Here is our one and only problem about Turing machines. The Turing machine $\mathcal{M}$ pictured in the figure recognizes the language $L \subseteq \{a,b\}^*$ consisting of all the strings in which the number of a's is *less than or equal to* the number of b's. As is our usual convention, if a transition for a given state and letter is not specified, we assume that the transition is to the reject state.

(a) Let Q be the set of the states of the machine and Γ the tape alphabet. Let

$$\delta : (Q - \{\text{accept}, \text{reject}\}) \times \Gamma \to Q \times \Gamma \times \{L, R\}$$

be the next-state function. What are the values of $\delta(1, X)$ and $\delta(1, a)$?

(b) Show the run of the machine (the complete configurations) on the inputs bab and a for ten steps, or until the machine halts, whichever comes first.

(c) Does $\mathcal{M}$ *decide* the language L?

(d) Is the language L decidable?

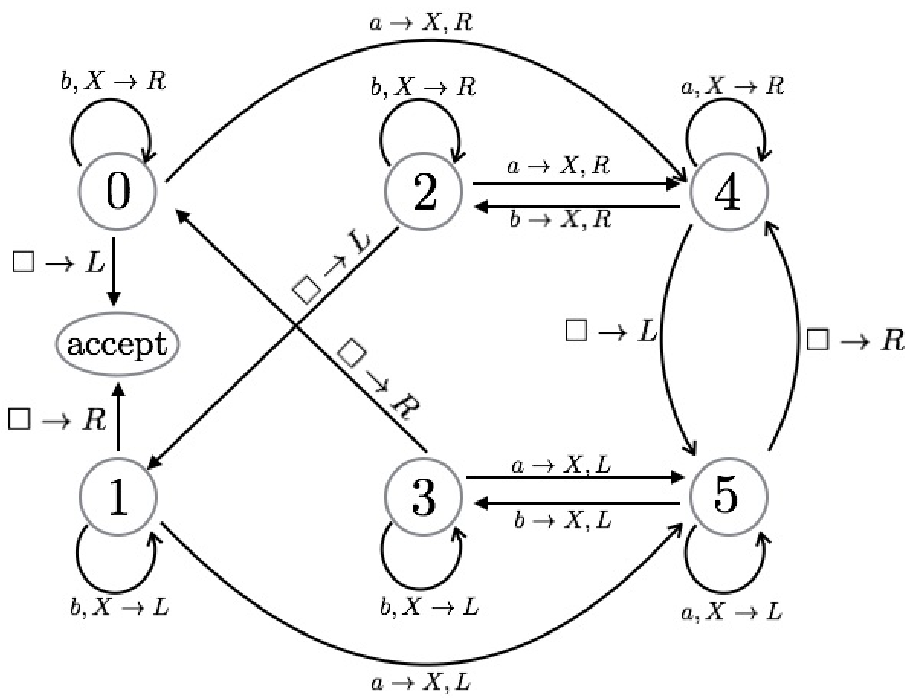

Figure 1: The Turing machine for Problem 4. The initial state is 0.