6.045J/18.400J: Automata, Computability and Complexity Prof. Nancy Lynch

Quiz 1

Please write your name in the upper corner of each page.

Problem 1:

True or False (20 points) Full credit will be given for correct answers. If you include justification for your answers, you may obtain partial credit for incorrect answers.

In all parts of this question, the alphabet Σ is $\{0, 1\}$.

1. True or False: A DFA with n states must accept at least one string of length greater than n.

2. True or False: A DFA with n states that accepts an infinite language must accept at least one string x such that $2n < |x| \leq 3n$.

3. If R is a regular language and L is some language, and $L \cup R$ is a regular language, then L must be a regular language.

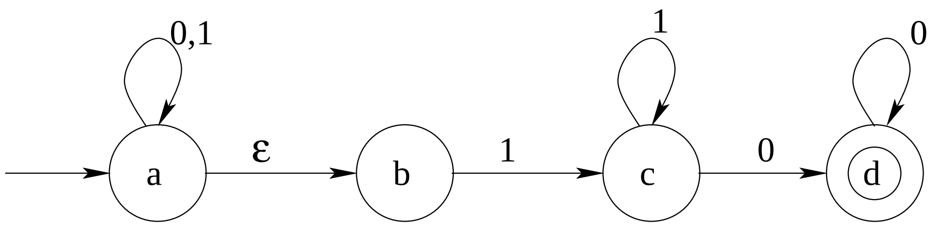
4. If F is a finite language and L is some language, and $L - F$ is a regular language, then L must be a regular language.

5. True or False: Define $FOUR(w)$, for a finite string w, to be the string consisting of the symbols of w in positions that are multiples of four. For example, $FOUR(1110011100) = 01$.
If L is a regular language, then $\{FOUR(w) : w \in L\}$ must be regular.

6. True or False: For every three regular expressions R, S, and T, the languages denoted by $R(S \cup T)$ and $(RS) \cup (RT)$ are the same.

7. True or False: If a language L is recognized by an n-state NFA, then it must be recognized by some DFA with no more than 2^n states.

8. If a language L is recognized by an 2^n-state DFA, then it must be recognized by some NFA with no more than n states.

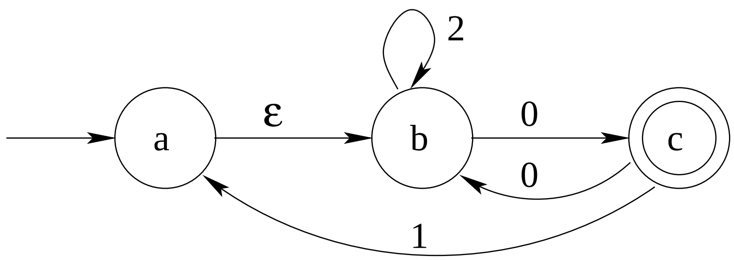


Problem 2: **(20 points)** Consider the following NFA:

1. **(12 points)** Convert this NFA into an equivalent DFA using the procedure we studied in class. **Your answer should be the state diagram of a DFA. Your diagram should include only the states that are reachable from the start state.** (Note: There are not more than 16 states in the resulting DFA). Please label your states in some meaningful way. You may explain your work, to receive partial credit for an incorrect answer.

2. **(4 points)** Give a regular expression that defines the language that is recognized by the given NFA (and therefore, also the DFA you constructed in part 1).

3. **(4 points)** Prove that there cannot exist a 2-state DFA for the language you defined in part (b) above. (Hint: Give three (short) strings that must lead to different states, *in any DFA that recognizes this language*.)



Problem 3: **(20 points)** Find a regular expression for the language recognized by this NFA M, using the procedure we studied in class. Remove the states in the order a, then b, then c.

Convert M to a g-NFA:

Remove state a:

Remove state b:

Remove state c:

Problem 4: **(20 points)** Describe a procedure (something that can be implemented using a program), that takes any two regular expressions and outputs a correct answer to the following questions:

1. (10 points) "Is $L(R_1) = L(R_2)$?"

2. (10 points) "Does $L(R_2)$ contain all the words in $L(R_1)$, plus exactly one additional word?"

Problem 5: **(20 points)**

If w is a string over an alphabet Σ and $\Sigma' \subseteq \Sigma$ is a (possibly) smaller alphabet, then we write $w|_{\Sigma'}$, the projection of w on Σ', for the string obtained from w by including just the symbols in Σ', that is, by removing all the symbols in $\Sigma - \Sigma'$.

For example, $012012012|_{\{0,1\}} = 010101$.

1. (14 points) Use the Pumping Lemma to prove that the following language L over the alphabet $\{0, 1\}$ is not regular:

$$L = \{wx : w, x \in \{0, 1, 2\}^* \text{ and } w|_{\{0,1\}} = (x|_{\{0,1\}})^R\}$$

   That is, the restriction of x to $\{0, 1\}$ is the reverse of the restriction of w to $\{0, 1\}$.

   For example, 0120122212010 is in L.

2. (6 points) Give an alternative proof that L is not regular based on a non-regularity result already proved in class or homework and one or more closure properties for regular languages.