CS 273, Fall 2007
Exam 1, September 25, 2007

INSTRUCTIONS (read carefully)

- Fill in your name and NETID below. Also write your netid on the other pages (in case they get separated).

NAME:

NETID:

- There are 6 problems, on pages numbered 1 through 6. Make sure you have a complete exam.
- The point value of each problem is indicated next to the problem, and in the table below.
- Points may be deducted for solutions which are correct but excessively complicated, hard to understand, or poorly explained.
- The exam is designed for one hour, but you have the full class period (75 min) to finish it.
- It is wise to skim all problems and point values first, to best plan your time.
- This is a closed book exam. No notes of any kind are allowed. Do all work in the space provided, using the backs of sheets if necessary. See the proctor if you need more paper.
- Please bring any apparent bugs to the attention of the proctors.
- After the midterm is over, discuss its contents with other CS 273 students **only** after verifying that they have also taken the exam (e.g. they aren't about to take the conflict exam).

Problem	Possible	Score
1	12	
2	8	
3	8	
4	8	
5	6	
6	8	
Total	50	

Problem 1: Short Answer (12 points, about 10 min)

The answers to these problems should be short and not complicated.

(a) List the elements of $\{2, 3\} \times \{a, b\}$.

(b) We've seen six operations that regular languages are closed under. List five of them. (Short names are sufficient.)

(c) $\mathbb{P}(\emptyset) =$

(d) Is it true that any DFA can be modified to create an equivalent DFA with only a single accept state?

(e) Suppose you have a DFA $M = (Q, \Sigma, \delta, q_0, F)$ recognizing a language L. Explain how to build a DFA M' which recognizes $\overline{L}$ (i.e. the set complement of L). One sentence should be sufficient.

(f) Is $(ab^* + ba^*)^* = (a + b)^*$?
If your answer is NO, give a string that is in one language but not in the other.

Problem 2: DFA design (8 points, about 12 min)

Let $\Sigma = \{a, b\}$. Let L be the set of strings that *do not* have the word aab as a (contiguous) substring

$$L = \{w \mid w \neq w_1 aab w_2 \ for\ any\ w_1, w_2 \in \Sigma^*\}$$

For example, $baa \in L$ and $abab \in L$, but $aab \notin L$, $babaabab \notin L$, and $ababaab \notin L$. Note that L contains all strings shorter than 3 characters.

Construct a DFA that accepts L and give a state diagram showing **all** states in the DFA. Your DFA should not contain more than 10 states.

Explain briefly how your DFA works. You do not need to prove formally that your DFA is correct. *You will receive* **zero** *credit if your DFA uses more than* **10** *states or makes significant use of non-determinism.*

Problem 3: NFA design (8 points, about 10 min)

Let $\Sigma = \{0, 1\}$. Let L be the language containing all strings over Σ that have length at least three and have a 0 in the third position from the end.

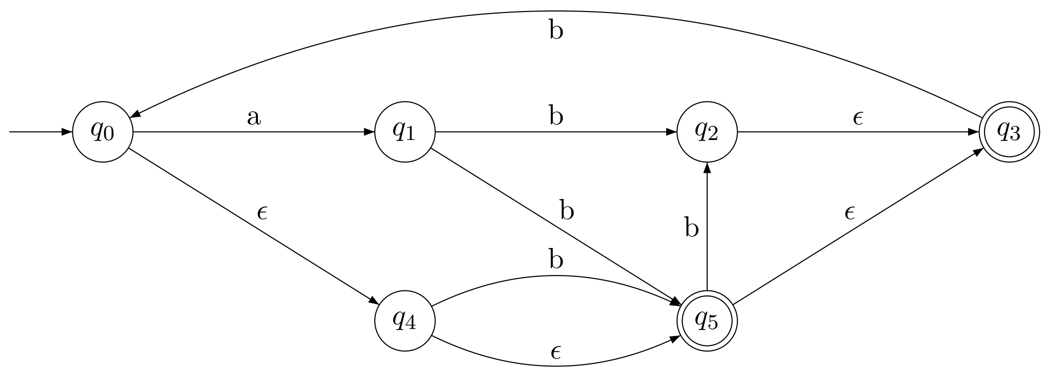
That is, $L = \{w_1 0 c_1 c_2 \mid w_1 \in \Sigma^*; \quad c_1, c_2 \in \Sigma\}$

For example, L contains 001000 and 11011, but L does not contain 00 nor 0100.

Give the state diagram of an NFA that recognizes L and has no more than 6 states. Full credit requires a machine that isn't overly complex and that exploits the extra features of NFAs. Don't solve this with a DFA.

Problem 4: NFA transitions (8 points, about 8 min)

Suppose that the NFA $N = (Q, \{a, b\}, \delta, q_0, F)$ is defined by the following state diagram:



Fill in the following values:

(a) $\delta(q_0, a) =$

(b) $\delta(q_1, b) =$

(c) $\delta(q_4, a) =$

(d) $\delta(q_2, b) =$

(e) $F =$

If S is any set of states, recall that $E(S)$, the ϵ-closure of S, is defined as

$E(S) = \{q \in Q |$ there is a state $r \in S$ such that q is reachable from r using zero or more ϵ transitions$\}$

Fill in the values of the following expressions:

(f) $E(\{q_0\}) =$

(g) $E(\{q_1, q_2\}) =$

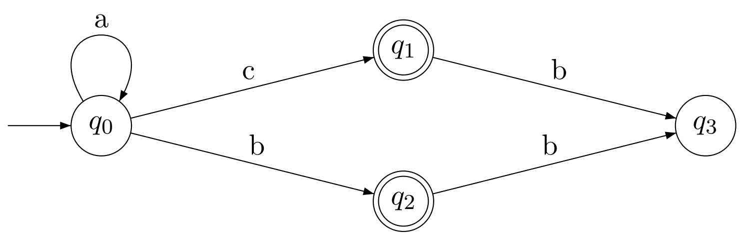
Regarding the language accepted by the NFA:

(h) Does the NFA accept the word bb? (Yes / No)

Problem 5: Short Answer II (6 points, about 8 min)

The answers to these problems should be short and not complicated.

(a) Here is the state diagram for an NFA recognizing some language L. Add extra states and/or transitions to this diagram, to produce the state diagram for an NFA recognizing L^*. (Do not remove states or transitions from the input NFA.)



(b) Suppose you have an NFA $M = (Q, \Sigma, \delta, q_0, F)$ which contains no ϵ transitions. We saw how to simulate M with a DFA $M' = (\mathbb{P}(Q), \Sigma, \delta', q_0', F')$. Suppose that S is a state of M' (i.e. $S \subseteq Q$), and $a \in \Sigma$. Then

$$\delta'(S, a) =$$

Problem 6: NFA modification and tuple notation (8 points, about 12 min)

Given a DFA M that accepts the language L, design a new NFA M' that accepts the language $L^R = \{w \mid w^R \in L\}$, where w^R is the reversed version of the string w. For example, if $w = aab$, then $w^R = baa$.

(a) Briefly explain the idea behind your construction, using English and/or pictures.

(b) Suppose that $M = (Q, \Sigma, \delta, q_0, F)$. Give the details of your construction of M', using tuple notation.