CS 4510 Automata and Complexity

Exam 1: Practice

- Name: ____ GTID: ____
- Any topic covered in lecture notes 1-6 and homeworks 1-3 are fair game for the exam. Absence of a topic from this practice exam does NOT imply an absence of that topic from the exam. Similarly, the actual exam may differ in length, format, and difficulty from this practice exam.
- We will go over this practice exam in class on Wednesday, Feb 9. In the meantime, you are encouraged to discuss this exam on Piazza and in office hours!
- Calculators are NOT permitted.
- You may use any of the theorems/facts/lemmas from the lecture notes, homeworks, or textbook without re-proving them unless explicitly stated otherwise.
- Good luck!

	Grade	Grading TA
Name on EACH page		
True/False		
Closure		
Regular Expressions		
Pumping Lemma		
Total		

Name: ____ GTID: ____

0. **Following Instructions** (0.5 points)

   Please write your name and GTID (letters, not numbers) at the top of EACH page of this exam.

1. **True or False** (4 points - 1 each)

   Circle one. No explanation necessary.

   (a) TRUE FALSE There is some language A for which A^* is empty.
   FALSE

   (b) TRUE FALSE Let D be a DFA with q states. Then there always exists some NFA N with q states such that $L(D) = L(N)$.
   TRUE

   (c) TRUE FALSE If a language obeys the pumping lemma, it is a regular language.
   FALSE

   (d) TRUE FALSE Let N be an NFA with states Q and final states F. Let N' be the same NFA but with final states $Q - F$. (In other words, form N' by swapping the accept and non-accept states of N.) Then $L(N') = \overline{L(N)}$. **FALSE**

Name: ______________________ GTID: ______________________

2. **Closure**

Here are three operations on languages. Also, let $L_1 = \{cat, 101\}$ and the alphabet be $\{c, a, t, 1, 0\}$.

- $FIRST\_OFF(L) = \{x \mid ax \in L \text{ for some } a \in \Sigma\}$. In other words, words in $FIRST\_OFF(L)$ are formed by removing the first letter of words in L, so that $FIRST\_OFF(L_1) = \{at, 01\}$.
- $END\_UP(L) = \{xa \mid x \in L, a \in \Sigma\}$. In other words, words in $END\_UP(L)$ are formed by adding an extra letter from the alphabet onto words in L, so that $END\_UP(L_1) = \{catc, cata, catt, cat0, cat1, 101c, 101a, 101t, 1010, 1011\}$.
- $SHIFT(L) = \{xa \mid ax \in L, a \in \Sigma\}$. That is, words in $SHIFT(L)$ are formed by taking a word from L and moving the first character onto the end, so that $SHIFT(L) = \{atc, 011\}$.

Prove that regular languages are closed under $SHIFT$, i.e., if L is regular then $SHIFT(L)$ is regular. (Hint: you might consider trying to prove closure for $FIRST\_OFF$ and END_UP first, and combining these ideas carefully to handle $SHIFT$).

Solution.

Let D be a DFA for L. To create D' from D for $SHIFT(L)$, we must do three things: skip the first transition of D, remember the character that we skipped, and add the skipped character onto the end. (Note that D is a DFA, but D' will be an NFA.)

To do so, we create one copy of D for each transition out of the start state. (Since D is a DFA, this just means we make one copy of D for each character in the alphabet.) Call these copies D_a, D_b, etc. for characters a, b, etc. in the alphabet.

We create a new start state s. We will transition from s to the "second" state of each copy. More explicitly, for each copy D_a, we use an ε-transition to go from s to the state in D_a corresponding to the a-transition from the old start state.

Within each copy, we transition as normal, but none of the final states are final states. Instead, we add an extra, accepting state outside of all of the copies called f, and we transition from each old final state in each copy to f using the character that copy is remembering. In other words, from the old final states of D_a, there is an a-transition to f.

You do not need to write your answer in formal math notation. However, for completeness, let $D = (Q, \Sigma, q_0, F, \delta)$. Then D' is the tuple $(Q', \Sigma, q_0', F', \delta')$:

- Let $Q_a = \{q_a \mid q \in Q\}$. Then $Q' = \{s\} \cup \{f\} \cup (\cup_{a \in \Sigma} Q_a)$.
- $q_0' = s$.
- $F' = \{f\}$.
-
  - $\delta'(s, \varepsilon) = \{r_a \mid r = \delta(q, a), a \in \Sigma\}$
  - for $q \in Q - F, a, b \in \Sigma$, if $\delta(q, a) = r$, then $\delta'(q_b, a) = \{r_b\}$.

– for $q \in F, a \neq b \in \Sigma$, if $\delta(q, a) = r$, then $\delta'(q_b, a) = \{r_b\}$ but $\delta(q_a, a) = \{r_a, f\}$.

Name: ____ GTID: ____

3. **Regular Expressions**

   Give a regular expression for the following languages. Please box your final answers.

   - The set of binary strings with at least one 0 and two 1's.

     Solution.

     $\Sigma^*0\Sigma^*1\Sigma^*1\Sigma^* \cup \Sigma^*1\Sigma^*0\Sigma^*1\Sigma^* \cup \Sigma^*1\Sigma^*1\Sigma^*0\Sigma^*$

   - The set of binary strings that alternate between 0's and 1's, so that each 1 not at the end of the string is followed by a 0, and each 0 not at the end of the string is followed by a 1. (In other words, the set of binary strings containing neither of 00 or 11 as substrings.)

     Solution.

     $(0 \cup \varepsilon)(10)^*(1 \cup \varepsilon)$

Name: ____ GTID: ____

4. **Pumping Lemma** (4 points)
Prove that the following language is not regular using the Pumping Lemma:

$$L = \{0^i 1^j 0^k \mid i, j, k \in \mathbb{Z}^{\geq 0}, \text{ exactly 2 of i, j, and k are equal}\}$$

Solution.

(a) Suppose, as a contradiction, that L is regular. Then let p be the pumping length.

(b) Pick the string $s = 0^p 1^p 0^{p-1}$. $|s| \geq p, s \in L$.

(c) According to the pumping lemma, s can be written as $s = xyz$ such that $|xy| \leq p$, $y \neq \varepsilon$, and $xy^i z \in L$ for non-negative integers i. Note that y must be some number of 0's, so we can write $y = 0^m$ for $1 \leq m \leq p$.

(d) According to the pumping lemma, $xyyz \in L$. But $xyyz = 0^{p+m} 1^p 0^{p-1} \notin L$ since $p + m > p > p - 1$. This is a contradiction. So our assumption was incorrect, and L is not regular.