CS 448/2405
Automata and Formal Languages
ASSIGNMENT # 1
NEW DUE DATE: Friday, October 3

1. Give deterministic finite automata accepting the following languages over $\{0, 1\}$.

   a. The set of all strings not containing the substring 111.

   b. The set of all strings of length at least four, and such that every block of four consecutive symbols contains at least 2 0's.

   c. The set of all strings with at least three symbols such that the third symbol from the right end is 1.

2. Prove of disprove the following for regular expressions r, s and t.

   a. $(rs + r)^* = r(sr + r)^*$ This language is not regular. When s is the empty string, then $(rs+r)^*$ equals r^* but this contains the empty string, whereas the right side does not.

   b. $s(rs + s)^* = rr^*s(rr^*s)^*$ This language is also not regular. Let r consist of the single string a and let s consist of the single string b. Then all strings in the language on the left begin with the symbol a, whereas all strings in the language on the right begin with the symbol b.

   c. $(r + s)^* = r^* + s^*$ This language is also not regular. Again, if $r = a$ and $s = b$, then ab is in the language on the left but not in the language on the right.

3. Are the following languages regular? Prove or disprove your answer.

   a. $L = \{w \mid w = w^R\}$ No, it is not regular. Suppose that it is an let M be a k-state DFA accepting L. Consider the string $w = 0^k1^k1^k0^k \in L$. Because M only has k states, it must loop (repeat states) on the first k symbols of s. That is, we can write $w = xyz$, where $x = 0^i$, $y = 0^j$, $z = 0^{k-i-j}1^k1^k0^k$, where $j \geq 1$ and $i + j \leq k$. Then the string xy^2z is also accepted by M. But this string is not in L, and thus we have reached a contradiction.

   b. $L = \{0^n \mid n \text{ is prime}\}$ Let k be the critical value given by the pumping lemma. Choose $s = 0^p$ where p is prime and $p > k$. (Such a prime exists since there are infinitely many primes.) Then s can be written as $0^l0^m0^n$, $m \geq 1$, $l + m + n = p$. Let

   $$s' = xy^{p+1}z = 0^l0^{(p+1)m}0^n = 0^{l+m+n+pm} = 0^{p(m+1)}.$$

   Then by the pumping lemma, s should also be in L if L is regular. But s' is not in L since $p(m + 1)$ is clearly not prime.

c. $L = \{0^n \mid n \text{ is even}\}$ $(00)^*$ is a regular expression accepting L.

d. $L = \{0^n 1^m 0^n \mid m, n \geq 0\}$ This is very similar to the example done in class. Let k be the critical value given by the pumping lemma. Then choose $s = 0^{p+1}1^{p+1}0^{p+1}$. This string is in L, so there exists i, j such that we can write s as $s = xyz$, where $x = 0^i$, $y = 0^j$ and z is the remainder of the string, and where $j \geq 1$, and $i + j \leq p$. Thus by the pumping lemma, the string xy^2z should also be in L, but s' is not, and hence we get a contradiction.

4. Sipser, Problem 1.41

5. Sipser, Problem 1.42

Let $M = (q_0, \Sigma, Q, \delta, F)$ be a DFA that accepts A. We will describe a nondeterministic automata, N, for accepting $A_{\frac{1}{2}-}$. On input x, N operates as follows. First, N guesses the state q^* that M would be in after it finishes reading x. Then in parallel (to be described later), N simulates M on x starting from state q_0, and N simulates M starting in state q^* on a guessed string y. To carry out this parallel computation nondeterministically (reading only x), the states of N will be triples of states $(q_*, q_i, q_j$ from Q. Let x be a string, $|x| = n$. Then after processing x (after n time steps), the set of states S_x that N can be in are as follows. A particular state (q_*, q_i, q_j) is in S_x if and only if: (1) M, when run on x beginning in q_0 ends in state q_i; and (2) There exists a string y, such that $|y| = |x|$ and such that M when run on y beginning in state q_*, ends up in state q_j. The final states F' of N will consist of triples (q_1, q_2, q_3) such that q_3 is a final state of M, and $q_1 = q_2$.

Assume that q_0 is the start state for M which is not a final state. The transition function for N is as follows. Let the start state be (q_0, q_0, q_0). First, nondeterministically transition to (q, q_0, q), for all $q \in Q$. (Here we are guessing that q will be the state that M would end up in after we process the string x.) Now let (q_*, q_i, q_j) be some state, and let the next symbol being read be b. Then (r_*, q_i', q_j') is in $\delta'(q_*, q_i, q_j)$ if and only if: $q_* = r_*$, and $\delta(q_i, b) = q_i'$, and finally, there exists a symbol q such that $\delta(q_j, a) = q_j'$.