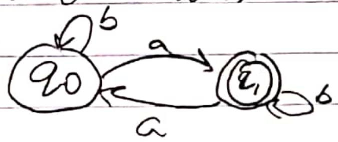
Max Shi

I pledge my honor that I have abided by the Stevens Honor System

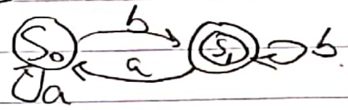
CS 334 Problem Set 2.

1a. w_1 = odd number of a's w_2 = ends with ab.

b. w_1 -

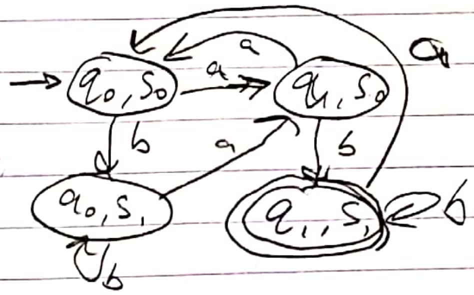


w_2 -

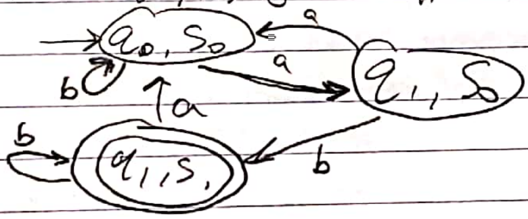


c

state	δ(state, a)	δ(state, b)
q_0, s_0	q_1, s_0	q_0, s_1
q_0, s_1	q_1, s_0	q_0, s_1
q_1, s_0	q_0, s_0	q_1, s_1
q_1, s_1	q_0, s_0	q_1, s_1

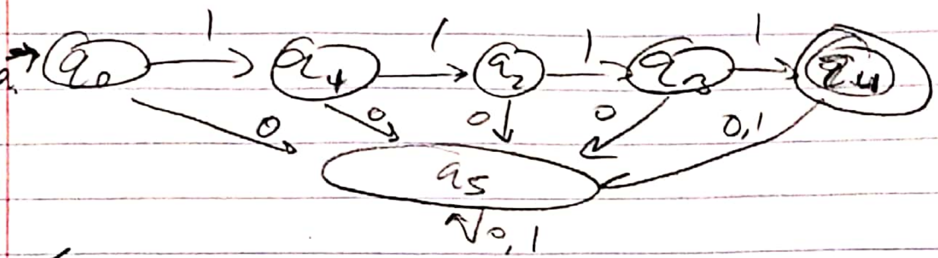


d. I can merge (q_0, s_0) and (q_0, s_1) into one state because they both have the same output for a and all b inputs from (q_0, s_0) get trapped in (q_0, s_1). Thus, they can be merged.



2. Let D be a DFA that accepts the language A. Thus A is regular and for all strings $w \in A$, the machine has a path from the start state to an accept state. If all arrows in D are reversed with the same input, with the old start state becoming the new singular accept state and with the new start state having empty transitions to each former accept state, which are no longer accept states, this new machine is now an NFA. This new NFA will accept all strings in A^R now because all strings w^R will be able to take the opposite path in the new NFA to the old start state. Then, because any NFA can be converted into a DFA, A^R is regular, and A being regular implies A^R is regular.

3a.



This cannot be reduced below 6 states because for q_n for $0 \le n \le 4$, q_n tracks the amount of 1's inputted into the machine. By this logic, q_0 is the start state with 0 1's and q_4 is the accept state for 4 1's. Furthermore, we need q_5 to catch all invalid inputs where the string 1111 is no longer possible. This sums to six states.

b. By the logic from 3a, let $L_n = \{$ strings of 1's of length $n \}$. The machine to accept this language requires states $q_0, q_1, \ldots q_{n-1}, q_n$ to track the number of 1's that have been inputted into the machine, and q_{n+1} to contain the dead state for when the string of 1's of length n is no longer possible. This amounts to $n+2$ states for a machine that accepts L_n, and because n starts at 1, this holds true for $k \ge 3$, as $n+2 \ge 3$. Thus, L_n can be accepted by a k-state FSA (where $k = n+2$) that cannot be recognized by any FSA with fewer states.