CSE491/596, Fall 2022 **Problem Set 1** **Due Thu. 9/22, 11:59pm**

(1) Let L_3 be the language of binary strings that represent *positive* multiples of 3. Define L to be the language of binary strings that *do not* have a substring in L_3. That is, L is the complement of $(0+1)^* \cdot L_3 \cdot (0+1)^*$. Follow these steps to characterize the language L.

(a) Design a DFA M_3 such that $L(M) = L_3$. It is possible that you might find this just in the course of innocently reading online notes, and that's OK, actually. (Just 3 pts.)

(b) Now make some small additions to your DFA M_3 to create an NFA N_3 such that $L(N_3) = (0+1)^* \cdot L_3 \cdot (0+1)^*$. (3 pts.)

(c) Convert N_3 into an equivalent DFA M' such that $L(M') = (0+1)^* \cdot L_3 \cdot (0+1)^*$. (12 pts.)

(d) Complement the final states of M' to get the needed M. (3 pts.)

(e) The string 11 sends M to a dead state wherever you start from because it is 3 in binary, so it makes a substring that belongs to L_3. Find a similar "dead substring" that does not have two consecutive 1s in it. (3 pts., for 24 total)

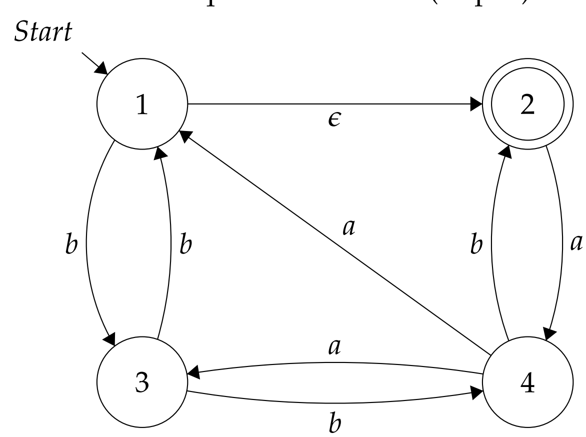
(2) Design both nondeterministic finite automata N_a, N_b, N_c and regular expressions r_a, r_b, r_c that denote the following three languages described in prose. (It is OK for your NFAs to have ϵ-arcs, and it is fine if one or more are DFAs since a DFA "Is-A" NFA.) All use the alphabet $\Sigma = \{0, 1\}$. $(3 \times (6+6) = 36$ points total)

(a) L_a = the set of binary strings in which the substring 10 occurs an odd number of times.

(b) L_b = the set of binary strings of the form $x = y00z$ where $|z|$ is odd.

(c) L_c = the set of binary strings having an occurrence of the substring 10 that do not have an occurrence of 11 after it.

(3) Convert the folowing NFA N into an equivalent DFA M (18 pts.):



Also answer (for $3 \times 3 = 9$ more pts., making 87 total on the set):

(a) Is there a string v that N cannot process from start to any state? Again give a shortest such v.

(b) Is there a string w such that no matter what string y follows it, the string wy is accepted? Again use your M.

(c) Stronger than (a) and counter to (b), is there a string z that N cannot process at all, not from any of its states p to any state q? Again use your M to explain your answer.