Question 2

Since A is regular, there is some DFA $M = (Q, \sum, \delta, q_0, F)$. Our goal is to construct a NFA that accepts A^R to show that A^R is regular. Let u be a state not in Q. Then consider the NFA $N = (Q \cup \{u\}, \sum, \delta_R, u, \{q_0\})$ where $\delta_R(u, \epsilon) = F$, and for any state v and letter a, $\delta_R(v, a) = \{q | q \in Q, \delta(q, a) = v\}$. So N is M but with all the arrows reversed, the inital state made into the only accepting state, and all of M's old accepting states are now an ϵ transition away from a new accepting state.

If the string $w = w_1 w_2 ... w_n$ is in A, than w corresponds to some path $q_0 q_{i_1} q_{i_2} ... q_{i_n}$ in the machine M where q_{i_n} is in F. If we put $w^R = w_n w_{n-1} ... w_1$ through N we can take the path $u q_{i_n} q_{i_{n-1}} ... q_{i_1} q_0$ because there is an ϵ transition from u to q_{i_n}, and all the transitions afterwards are the reverse of transitions in N. Since q_0 is a final state in N, N accepts w^R. This means N accepts the revers of any word in A, so $A^R \subseteq L(N)$. Notice we do not know if N accepts more than just A^R.

Now suppose $w = w_n w_{n-1} ... w_1$ is in $L(N)$. Then if we put w through N we can take the path $u q_{i_n} q_{i_{n-1}} ... q_{i_1} q_0$ where q_{i_n} is in F. Then, since M is N with transitions reversed, string w^R corresponds to path $q_0 q_{i_1} q_{i_2} ... q_{i_n}$ in M. So M accepts w^R because q_{i_n} is in F. This implies that any word in $L(N)$ is the reverse of a word in A. Thus $L(N) \subseteq A^R$ and so we have $L(N) = A^R$. We know now N is a NFA accepting A^R, and so A^R is regular.

Question 3

(a) We can create a DFA M with k states that accepts L. Let M' be exactly the same as M except all the accepting states in M are nonaccepting states in M', and all the nonaccepting states in M are accepting states in M'. If M accepts string w, w must have correspond to a path ending in an accepting state. So if we put w through M', we end at the same state except it is nonaccepting. So any string in L is not in $L(M')$. On the other hand, suppose string v is not in L. Then if we put string v through M it must end at a nonaccepting state. So putting v through M' we end at the same state which is now accepting. So any string not in L is accepted by M'. This shows that some DFA with k states, M', accepts $\overline{L}$.

Suppose, contrary to what we wish to show, there exists a DFA N over j states with $j < k$ that accepts $\overline{L}$. Then from the argument above, we can change the nonaccepting states to accepting states and vice versa in N to create a new machine N' with j states that accepts $\overline{\overline{L}} = L$. This contradicts thats that k was the minimal amount of states in a machine that recognizes L! So k is the minimal amount of states in a machine that recognizes $\overline{L}$.

(b) Consider the set $S = \{\epsilon, a, aa, ..., a^i, a^{i+1}\}$. Now let $0 \le j < k \le i + 1$. Then $a^j a^{i-j} = a^i \in L$, and since $i + k - j > i$ we have $a^k a^{i-j} = a^{i+k-j} \notin L$. So a^j and a^k are distinguishable, showing that any pair of strings in S are distinguishable. Notice also that a^l and a^m for $l, m > i + 1$ are not distinguishable since appending any string to either a^l or a^m will create a string of length greater than i which consequently would not be in L. So we may include at most one string of length greater than i in any set of pairwise distinguishable strings. S includes one string of length greater than i as well as all strings of length less than or equal to i and so it must be maximal. The index of L is then $|S| = i+2$, and the Myhill-Nerode Theorem, this is minimal number of states in a DFA that accepts L.

(c) Since $\overline{L}$ us finite, let a^i be the longest string in $\overline{L}$. By part b, $\overline{L}$ is regular and the minimal number of states in a DFA that accepts it is $i+2$. So by part a, L is regular as well and the minimal number of states in a DFA that accepts L is $i+2$.

(d) Since the longest string in L is of length 7, by part b the machine with fewest states that recognizes it has $7+2=9$ states.

From class, we can recall that $a^2 3$ is the longest number not in $L*$. We can also check this fact though. We can make a set S of the first few shortest strings in $L*$ step by step. At step 0, we put a^0 in S. And then at step i for $i > 0$, we place a^i in the list if a^{i-5} or a^{i-7} was already in the list. So at step 7, $S = \{\epsilon, a^5, a^7\}$. At step 28, $S = \{\epsilon, a^5, a^7, a^{10}, a^{14}, a^{15}, a^{17}, a^{19}, a^{20}, a^{21}, a^{22}, a^{24}, a^{25}, a^{26}, a^{27}, a^{28}\}$. So $a^2 3$ is not in L. We can also see inductively from S at this step that at step $k > 28$, a^{k-5} will already be in S. So every string of length greater than 23 is in L.

This implies that a^{23} is the longest string in $\overline{L}$. So by part b, the minimum number of states for a DFA recognizing $\overline{L}$ is $23+2=25$, and part a then says that the minimum number of states for a DFA recognizing L is 25 as well.

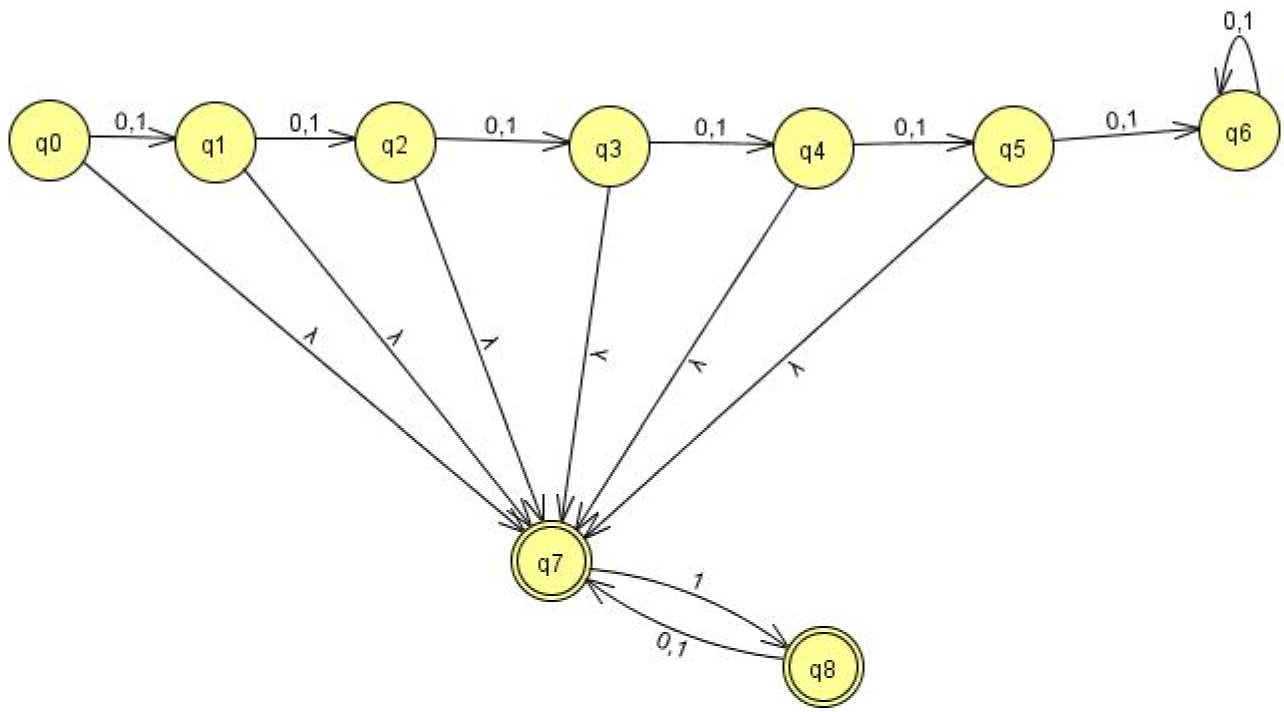

Figure 1: An answer to Problem 1

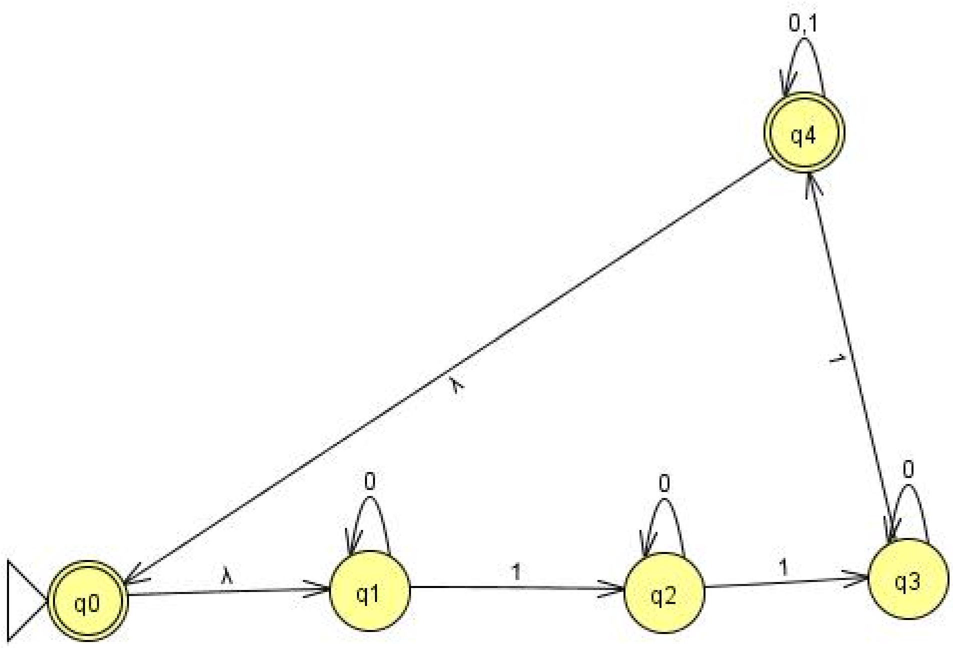

Figure 2: An answer to problem 2.a.

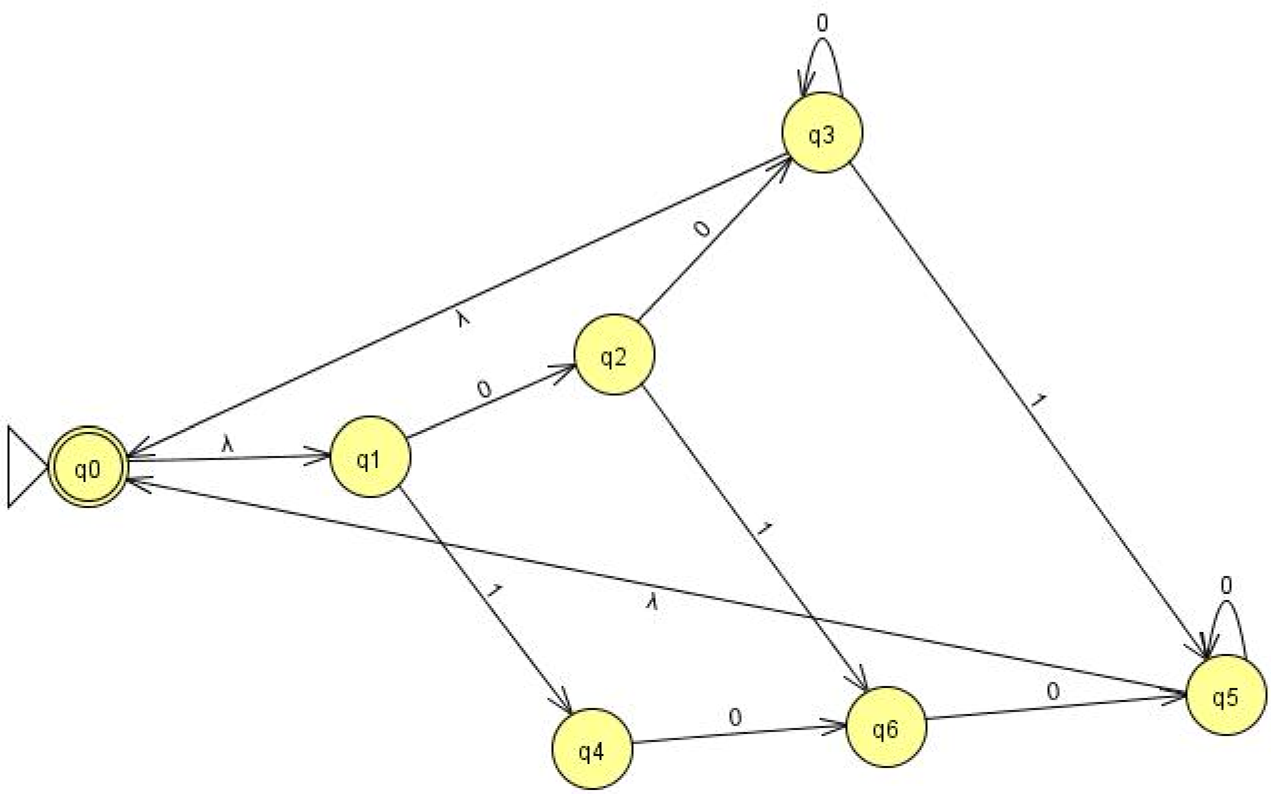

Figure 3: An answer to problem 2.b.