Models of Computation (Sheet #2)

Marius Gavrilescu

1. (a) `b*(ab*a)b*`
   (b) `a*(ba*b)a*ba*`
   (c) `b*(ab*a)b*|a*(ba*b)a*ba*`

3. $E_{q_0,q_0}^{\{q_0,q_1,q_2\}} = E_{q_0,q_0}^{\{q_0,q_1\}} + E_{q_0,q_2}^{\{q_0,q_1\}}(E_{q_2,q_2}^{\{q_0,q_1\}})^* E_{q_2,q_0}^{\{q_0,q_1\}} = \varepsilon + (a^*ba)(aa^*ba)^*(aa*) = a^*ba(aa^*ba)^*aa^*$.

4. (a) False. `EF` matches the LHS, does not match the RHS.
   (b) We will use induction.

   Base case: `(EF+E)*` is expanded 0 times. In this case we have `E=E(FE+E)*`, true.

   Suppose $(EF+E)^k E \equiv E(FE+E)^*$. Then $(EF+E)(EF+E)^k E = E(F+\varepsilon)E(FE+E)^t = E(FE+E)(FE+E)^t = E(FE+E)^{t+1} \equiv E(FE+E)^*$.

   By induction we get the desired result.

   (c) We have $(a+b)^* = a^*(ba^*)^*$. Applying this to the LHS we get $E(FE+E)^*F = EE^*(FEE^*)^*F$.

   We will now use induction. If $(FEE^*)^*$ is expanded 0 times, then
   $EE^*F \equiv EE^*F(EE^*F)^*$, true.

   Suppose $EE^*(FEE^*)^kF \equiv EE^*F(EE^*F)^*$.
   Then $EE^*(FEE^*)(FEE^*)^kF = EE^*FE^*E(FEE^*)^kF = EE^*FE^*EE^*F(EE^*F)^t = (EE^*F)(EE^*E^*F)(EE^*F)^t = EE^*F(EE^*F)^{t+1}$.

   By induction we get the desired result.

5. Take a DFA accepting L. Reverse all edges. The only accepting state in the new NFA will be the original starting state. The new start state will be a new state that has epsilon transitions to all accepting states of the original automaton.

6. (a) Take a DFA accepting L. Remove all outgoing edges from every accepting state. Now the new NFA will accept only states that are in L (because we did not add accepting states or edges), and won't accept any string which has a proper prefix in L.
   (b) Take a DFA accepting L. For every accepting state S, do a graph search from this state, searching for other accepting states. If an accepting state is found, S is removed from the set of accepting states.

   The new DFA will accept only states that are in L (because all we did was turn some accepting states into non-accepting ones), and it will not accept any string that is a proper prefix of an accepted string.

   (c) Take a DFA accepting L. For every state of the automaton, mark it if the automaton would accept x if started from that state. The set of marked states is the new set of accepting states.