CSC236 tutorial exercises, Week #11
best before Friday afternoon

1. Recall that $\emptyset$ is a valid (but seldom used) symbol in our regular expression syntax. Construct an NFSA M for the regular expression $R = \emptyset a$, using the procedure described in section 7.6.1 of the course notes. Does the result make sense with the meaning of R?

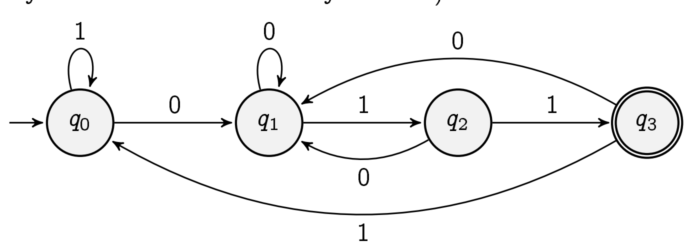
2. Several of the closure results from this week's readings become a bit simpler to prove if we assume that our FSA has only one accepting state. In this question, you will prove that this is something we can assume without loss of generality.

   (a) Show that for any FSA M, there exists an NFSA M' having only one accepting state, such that $\mathcal{L}(M) = \mathcal{L}(M')$. (You may use graphics to illustrate your construction, as in the Vassos notes, or describe it in words.)

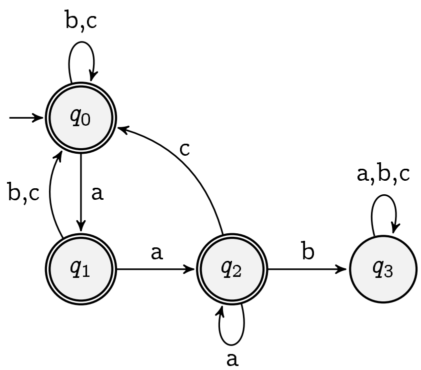
   (b) Is it also true that there always exists an equivalent DFSA with only one accepting state? If yes, give a constructive procedure for creating such a DFSA. If not, find a counterexample (and try to argue why it has no equivalent DFSA with a single accepting state).

3. For a language L, define $\text{Trim}(L)$ to be the language resulting from truncating the last symbol from each string in L. Formally $\text{Trim}(L) = \{x \in \Sigma^* \mid \exists w \in L, a \in \Sigma, w = xa\}$. In this question, you will show that the set of languages recognized by FSAs is closed under 'trimming'.

   (a) Modify the following DFSA M to form a new FSA M' such that $\mathcal{L}(M') = \text{Trim}(\mathcal{L}(M))$. (You may use non-determinism if you wish.)



   (b) Repeat the procedure above for the following DFSA:



(c) Describe a general procedure for constructing an FSA that accepts $\mathrm{Trim}(\mathcal{L}(M))$ given an arbitrary DFSA M.