Reading assignment: Sipser’s book, sections 1.2 and 1.3.

Instructions: Same as for Problem set 1.

1. Exercise 1.10, Sipser’s textbook. (Swapping accept and non-accept states in DFAs and NFAs)

2. Recall the subset construction converting NFAs to DFAs that we discussed in class. If $N$ was the original NFA with start state $s$, and $\Delta, s'$ were the transition function and start state respectively of the constructed DFA $D$, give a formal proof by induction of the following claim (which we used to argue that the construction worked):

   \[ \text{For all } x \in \Sigma^*, \hat{\Delta}(s', x) = \text{Reach}_N(s, x). \]

3. Use the subset construction to convert the NFA in Exercise 1.12 (b) of Sipser’s book to an equivalent DFA.

4. (20 points) Construct an NFA (giving your answer in form of a state diagram) as well as a regular expression for each of the following languages over the alphabet \{0, 1\}:

   (a) $L_1 = \{ w \mid w \text{ is any string except 11 and 111}\}$.
   (b) $L_2 = \{ w \mid w \text{ contains an equal number of occurrences of the substrings 01 and 10}\}$.
      (Thus 101 $\in L_2$ because 101 contains a single 01 and a single 10, but 1010 $\notin L_2$ because 1010 contains two 10s and one 01.)
   (c) $L_3 = \{ w \mid w \text{ contains two 0's that are separated by a number of positions that is a multiple of 3}\}$. (Note that 0 is also a multiple of 3.)

5. Suppose that $L$ is regular, i.e. is recognized by a DFA. Then:

   (a) Prove that $L^R = \{ w^R \mid w \in L\}$ is also regular. (Here $w^R$ denotes the reverse of the string $w$, i.e. if $w = w_1w_2, \ldots, w_n$ then $w^R = w_nw_{n-1}\ldots w_2w_1$.)
   (b) Prove that $\text{MIN}(L) = \{ w \in L \mid \text{no proper prefix of } w \text{ is in } L\}$ is also regular.
   (c) * (Extra Credit) Define $\text{HALF}(L) = \{ x \mid \text{for some } y \text{ such that } |x| = |y|, xy \text{ belongs to } L\}$. Prove that $\text{HALF}(L)$ is regular if $L$ is regular.
      (For all of the parts above, presenting a correct construction of an NFA/DFA for the languages in question, along with an informal yet convincing explanation of why it does the right thing, will suffice. Fully formal proofs by induction are not required.)