1. This problem consists of four parts, three of which are designed to improve your understanding of the Myhill-Nerode Theorem and the DFA minimization algorithm.

   (a) (*) Recall the following DFA $M$, which we saw in Homework 1:

   ![DFA Diagram]

   Determine the equivalence relation $R_M$ for this DFA. You should specify the equivalence relation by writing down each of its equivalence classes in the form of a regular expression. [NOTES: (i) Recall that the relation $R_M$ is defined by $xR_M y$ iff $M$ ends up in the same state on inputs $x$ and $y$. (ii) Recall also from HW1 that you have already described the set of strings that take $M$ to each of its states.]

   (b) (*) In class we showed that if $L$ is a regular language, then the equivalence relation $R_L$ (indistinguishability) has only finitely many equivalence classes (these correspond to the states of the minimal DFA for $L$). Now consider the language $L = \{0^n1^n : n \geq 0\}$. By describing the equivalence classes of $R_L$, prove that $L$ is not regular. [NOTES: (i) Recall that the relation $R_L$ is defined by $xR_L y$ iff $\forall z (xz \in L \iff yz \in L)$. Of course, we could also prove that $L$ is not regular using the pumping lemma; the point of this problem is to get you to use a different method based on the Myhill-Nerode Theorem.]

   (c) Now use the pumping lemma to show that $L$ is not regular.

   (d) (*) Apply the minimization procedure discussed in class to construct a minimal DFA that is equivalent to the following DFA. Show clearly the steps you used to arrive at your answer; you should consider the pairs of states in lexicographic order.

   ![Minimized DFA Diagram]
2. Consider the language \( L = \{ w = a^i b^j c^k : i, j, k \geq 0 \text{ and if } i = 1 \text{ then } j = k \} \).

(a) Show that \( L \) "acts like" a regular language with respect to the pumping lemma. Specifically, give a pumping length \( p \) and show that \( L \) satisfies the conditions of the lemma for this \( p \).

(b) Now show that \( L \) is NOT regular.

(c) Why is this not a contradiction?

3. (*) Show that for any positive integer \( n \) there is a language \( L_n \) for which both of the following statements hold

(a) There is a DFA with \( n \) states that recognizes \( L_n \) and

(b) No DFA with fewer than \( n \) states recognizes \( L_n \).

4. The pumping lemma says that every regular language \( L \) has a pumping length \( p \), and that any string \( w = w_1 w_2 \cdots w_n \in L \) with \( n \geq p \) can be pumped. Clearly if \( p \) is a pumping length for \( L \), so is \( p' \) with \( p' \geq p \). The minimum pumping length for \( L \) is the smallest \( p \) that is a pumping length for \( L \). [e.g., when \( L = 01^* \), the minimum pumping length is 2: the string \( w = 0 \) is in \( L \), has length 1, but cannot be pumped (why?), but any string in \( L \) of length \( \geq 2 \) must contain a 1 and can be pumped (why?, how?).]

For each of the following languages, find the minimum pumping length and justify your answer.

(a) \( L = 0001^* \)
(b) (*) \( L = 1011 \)
(c) \( L = 0^*1^* \)
(d) (*) \( L = 001 \cup 0^*1^* \)
(e) (*) \( L = 10(11^*0)0 \)
(f) \( L = 0^*1^*0^*1^* \cup 10^*1 \)
(g) (*) \( L = (01)^* \)
(h) (*) \( L = \epsilon \)
(i) (*) \( L = 1^*01^*01^* \)