Problem 1. (75 points) Consider the regular expression \( R = (0 \cup 10 \cup 100)^* \).

a. Draw a nondeterministic finite automaton \( N \) such that \( L(N) = L(R) \). (You may take shortcuts that omit \( \varepsilon \) transitions.)

b. Use the subset construction to draw a deterministic finite automaton \( D \) such that \( L(D) = L(N) \). (Label each state of \( D \) with the corresponding set of states of \( N \).)

c. Use the minimization algorithm followed by the quotient construction to draw a minimal DFA \( M \) such that \( L(M) = L(D) \). (Label each state of \( M \) with the corresponding equivalence class of states of \( D \).)

d. What is the index of \( L(R) \)? If the index of \( L(R) \) is \( k \), give \( k \) words \( w_1, \ldots, w_k \) such that no two of the words are \( \equiv_{L(R)} \) equivalent. (You need not justify your answers.)

e. Give a regular expression \( S \) such that \( L(S) = [1]_{\equiv_{L(R)}} \). (Here \([1]_{\equiv_{L(R)}} \) stands for the \( \equiv_{L(R)} \) equivalence class of the one-letter word \( 1 \).)

Problem 2. (75 points) For every language \( A \subseteq \Sigma^* \), we define the following two languages:

\[
\text{double1}(A) = \{x_1 x_2 x_3 \ldots x_{2n} \in \Sigma^* \mid n \geq 0 \text{ and } x_1 x_2 x_3 \ldots x_n \in A\}
\]

\[
\text{double2}(A) = \{x_1 x_2 x_3 \ldots x_{2n} \in \Sigma^* \mid n \geq 0 \text{ and } x_1 x_3 x_5 \ldots x_{2n-1} \in A\}
\]

a. For \( \Sigma = \{0, 1\} \) and \( B = L(0^*) \), describe in words the two languages \( \text{double1}(B) \) and \( \text{double2}(B) \).

b. Which of the following two statements are true and which are false?

S1 If \( A \) is a regular language, then \( \text{double1}(A) \) is also regular.

S2 If \( A \) is a regular language, then \( \text{double2}(A) \) is also regular.

To argue that one of these statements is true, you must define a finite automaton \( M' = (Q', \Sigma', \delta', q_0', F') \) that accepts \( \text{double}(A) \) from a given finite automaton \( M = (Q, \Sigma, \delta, q_0, F) \) that accepts \( A \).

To argue that one of these statements is false, you must find a regular language \( A \) and prove, using the pumping lemma, that \( \text{double}(A) \) is not regular.