Here and later when you are asked to give an automaton, you should give it as a state diagram (as in Problem 1(b) below), unless otherwise stated.

1. Give a DFA for each of the following languages over the alphabet \{ a, b, c \}:
   
   (a) strings ending in abc
   (b) strings beginning abc
   (c) strings that do not contain the substring ab
   (d) strings where each symbol in an odd-numbered position is b
   (e) strings where the difference between the number of a’s and the number of b’s is odd.

2. (a) Let \( A = \{ a \} \) and \( B = \{ b, c \} \). List the first ten elements of the language \((A \circ B)^*\) in lexicographic order.

   (b) Describe in English the languages recognised by the following DFAs:

3. Combine the below two DFAs into a DFA for the union of their languages, following the construction given in the textbook (Theorem 1.25). Show the computation of the union automaton on input 0110.

Continues on the next page!
4. (a) Assume that the languages \( A \) and \( B \) are regular. Prove that also their intersection \( A \cap B \) is regular by suitably modifying the proof of Theorem 1.25 in Sipser’s book.

(b) Show that if a language \( A \) is regular, then so is its complement \( \overline{A} \).

(c) Give an alternative proof for the result in part (a) by applying the results of part (b) and Theorem 1.25.

5. Show that for any language \( A \) we have \( A = A^* \) if and only if \( AA \subseteq A \) and \( \varepsilon \in A \).

6. [Sipser 1.42] The perfect shuffle of two languages \( A \) and \( B \) is defined as

\[
\{ a_1b_1a_2b_2\ldots a_nb_n \mid n \in \mathbb{N}, a_i, b_i \in \Sigma \text{ for all } 1 \leq i \leq n, a_1\ldots a_n \in A \text{ and } b_1\ldots b_n \in B \}.
\]

Show that the class of regular languages is closed under perfect shuffle. In other words, if \( A \) and \( B \) are regular, then so is their perfect shuffle.