Homework problems:

1. Give regular expressions describing the following languages:
   (a) \( \{ w \in \{a, b\}^* | w \text{ contains } abb \text{ as a substring} \} \);
   (b) \( \{ w \in \{a, b\}^* | w \text{ contains either } abb \text{ or } bba \text{ (or both) as a substring} \} \);
   (c) \( \{ w \in \{0, 1\}^* | w \text{ contains exactly two } 0's \} \);
   (d) \( \{ w \in \{0, 1\}^* | w \text{ contains at least two } 0's \} \);
   (e) \( \{ w \in \{0, 1\}^* | w \text{ contains an even number (possibly zero) of } 0's \} \);
   (f) \( \{ w \in \{0, 1\}^* | w \text{ begins and ends with different symbols} \} \);
   (g) \( \{ w \in \{0, 1\}^* | |w| = 1 \mod 3 \} \).

2. (a) Construct in a systematic way (as described in your textbook) a nondeterministic
    finite automaton corresponding to the regular expression \((\varepsilon \cup 0)^*011^*\).
    (b) Make your automaton deterministic.
    (c) Describe the language in part (a) in English as simply as you can.

3. Give regular expressions describing the following languages:
   (a) \( \{ w \in \{a, b\}^* | w \text{ does not contain } abb \text{ as a substring} \} \);
   (b) \( \{ w \in \{0, 1\}^* | w \text{ contains an even number of both } 0's \text{ and } 1's \} \).
   (Hint: Design first a finite automaton for each of the languages, cf. problems 2/3(c)
   & 3/1, and convert these automata then in a systematic manner, as described in your
   textbook, into the corresponding regular expressions.)

Demonstration problems:

4. Simplify the following regular expressions (i.e., design simpler expressions describing
   the same languages):
   (a) \( (\emptyset^* \cup a) (a^*)^* (b \cup a) b^* \)
   (b) \( (a \cup b)^* \cup \emptyset \cup (a \cup b)b^*a^* \)
   (c) \( (b^* \cup a^*) (a^*b^*)^* \)

5. Determine whether the regular expressions \( r_1 = b^*a(a^*b^*)^* \) and \( r_2 = (a \cup b)^*a(a \cup b)^* \)
   describe the same language, by constructing the minimal deterministic finite automata
   corresponding to them.

6. Prove that if \( L \) is a regular language, then so is \( L' = \{ xy | x \in L, y \notin L \} \).