1. Read Kozen Lectures 1 - 4.

2. Give a deterministic finite automaton (DFA) for each of the following sets (a graph representation is sufficient):

(a) the set of strings in \( \{a, b\}^* \) containing a substring of four consecutive b's;

(b) the set of strings in \( \{a\}^* \) whose length is divisible by either 3 or 5;

(c) the set of strings in \( \{0, 1\}^* \) containing an odd number of 0s and an even number of 1s;

(d) the set of strings over \( \{a, b\} \) containing at least 2 occurrences of 3 consecutive a's, overlapping permitted (e.g., the string aaaa should be accepted).

(e) the set of strings in \( \{0, 1, 2\}^* \) that are ternary (base 3) representations, leading zeros permitted, of numbers that are not multiples of four. (Consider the null string a representation of zero.)

3. Let \( M = (Q, \Sigma, \delta, s, F) \) be a DFA. Prove that for all input strings \( x, y \) in \( \Sigma^* \) and states \( q \) in \( Q \),

\[
\hat{\delta}(q, xy) = \hat{\delta}(\hat{\delta}(q, x), y).
\]

(Hint: Use induction on \( |y| \). You will need to use the definition of \( \hat{\delta} \).)

4. Regular expressions are a useful tool in a number of domains. In a short paragraph, describe in your own words the applicability of regular expressions. Give 2 specific examples of regular expression usage. One of your examples must be from a programming language that supports regular expressions – name the language, give an example regular expression, and what it achieves. The other example can be from a command-line or search bar, e.g., file name pattern matching.