Exercise 1. For each of the following REs on the alphabet $\Sigma = [a, b, c]$, identify one word that is **in the language** recognized by the RE, and one word that is **not in the language**.

(a) $(ab)^* + bc^*$

Solution:
Your solution Here

(b) a^*bc^*

Solution:
Your solution Here

(c) $(a + b)^*(b + c)^*$

Solution:
Your solution Here

Exercise 2. Give a regular expression for each of the following languages. In all cases, the alphabet is $\Sigma = \{0, 1\}$.

(a) $\{w \in \Sigma^* \mid w$ `has an odd number of 0s followed by a single 1`$\}$

Solution: Your solution Here

(b) $\{w \in \Sigma^* \mid w$ `starts in a triple letter (000 or 111)`$\}$

Solution: Your solution Here

(c) $\{w \in \Sigma^* \mid w$ `contains exactly three 1s or at least two 0s`$\}$

Solution: Your solution Here

Exercise 3. Show that any language L_F containing only finitely many strings is regular.

Solution: HINT: To prove L_F is regular, we simply need to show there is a regular expression to generate it;

Your solution Here □

Exercise 4. Show that if L is a regular language, then the reverse language L^R:

$$L^R = \{w^R \mid w \in L \texttt{ and } w^R \texttt{ is the word } w \texttt{ written in reverse}\}$$

is also a regular language (i.e. regular languages are closed under **reversal**).

Remember, proving a language is regular is all about proving it has a regular expression that can generate it!

Hint: try induction! by:

1. **Base cases**: prove this works for the 3 base cases:
   - $L = \emptyset$ (show the reverse of an empty language is regular)
   - $w = \epsilon$ (show you can get the regular expression for the reverse of an empty word)
   - $w = a$ where a represents any single valid symbol(show you can get the regular expression for the reverse of a word made of a single symbol)
2. **Induction Hypothesis**: assume that
   - the alternation of two regular expressions is also a regular expression
   - the concatenation of two regular expressions is also a regular expression
   - applying Klenee Star to a regular expression results in a regular expression
3. **Induction Step**: Using the hypotheses, show that the reverse of any regular expression is also a regular expression (the reverse of regular expressions using alternation, concatenation, and Kleene star are also regular expressions);

Solution: Your solution Here

References

[1] Sipser, Michael. *Introduction to the Theory of Computation.* Course Technology, 2005. ISBN: 9780534950972

[2] Critchlow, Carol and Eck, David *Foundation of computation.*, Critchlow Carol, 2011