Sipser, Section 1.3: Regular Expressions

Reading: All of section 1.3.

Homework Assignment 7
Due: November 16, in class

Homeworks are due at the beginning of class on the due date. Late homeworks will not be graded for credit, but I will give comments and feedback on them.

Remember to read the descriptions carefully, and think about things that are not mentioned: e.g., if a string is described as containing at most one 1, it may contain no 1s at all; it may also contain any number of 0s!

1. Do Exercise 1.15 from Sipser. To help me with correcting, please write each regular expression along with your answers.

2. Draw an NFA for each of the following regular expressions. (Again, please give the regular expression along with your answer).
   a. 010*1
   b. (010*1)*
   c. (010*1)*1
   d. (010*1)*1(0∪1)*
   e. 1(010*1)*
   f. 1(010*1)*(0∪1)* (Hint: this one can be done by a simpler automaton than any of the others)

3. a. Give a regular expression that matches the set of all strings containing a total of exactly two 1s (i.e., any number of 0s are allowed).
   b. Give a regular expression that matches any string containing the substring 010 anywhere within them.

4. Do Exercise 1.13 from Sipser, except for part f.
   The following directions should make the problems easier. Ignore them at your peril!
   • You could use the GNFA construction (from Lemma 1.32) to solve these problems, but it is a lot simpler to write the regular expressions from scratch.
   • Part h is somewhat messy, but not hard:
     (a) Start by writing a regular expression that matches all acceptable strings of length 2, then of length 3.
     (b) Now write a regular expression that matches all strings of other lengths.
(c) Now solve the problem.

• For part j, the trick is to examine where in the string the 1 could appear, in relation to the first two 0s (there could be more than two!)
  (a) First write a regular expression for a string with at least two 0s, that consists only of zeros.
  (b) Now consider the possibilities for the 1:
     i. There could be no 1 in the string.
     ii. The 1 is before the first 0.
     iii. The 1 is between the first and second 0.
     iv. The 1 is somewhere after the second 0.
     Write a regular expression to cover each case; e.g., write a regular expression for a string with at least two 0s no 1s, another for a string with at least two 0s and a single 1 before all of them, etc.
  (c) Now answer part j by combining the possibilities in (b).

• Build up to part l by writing regular expressions for the following types of strings. Each type builds on the previous ones, so do them in order:
  (a) Strings made exclusively of 0s, with an even number of 0s.
  (b) Strings containing exactly two 0s (plus any amount of 1s anywhere).
  (c) Strings containing an even number of 0s. (Again, 1s are allowed).
  (d) The string consisting of two 1s and nothing else.
  (e) Strings containing exactly two 1s.
  (f) Now go back and answer part l!

5. Writing the DFA for Exercise 1.4 f was relatively easy, provided one knows the necessary trick. But writing a regular expression for it is hard; you’d probably need to use the GNFA construction to do it.

You are not required to solve part f; instead, take a good look at it and explain what it is about this problem that makes it difficult to write a regular expression for it.

6. Optional: Do Exercise 1.16 from Sipser.