Finite-State Automata, Regular Expressions, and Grammars

For solutions to these problems, please visit `www.vincematsko.com` and click on Riga 2010.

1. Write a regular expression for the language accepted by the following grammar. In this problem and others, V represents the set of symbols, S the set of terminal symbols, and $V \setminus S$ the set of non-terminal symbols. Any string in the language must consist entirely of terminal symbols. Thus, rules must be applied until no non-terminal symbols remain.

   $G = (V, S, v_0, \mapsto)$

   $V = \{v_0, v_1, v_2, a, b\}, \quad S = \{a, b\}$

   $$\begin{array}{llr} \mapsto: & v_0 \mapsto av_1 & (1) \\ & v_1 \mapsto bv_2 & (2) \\ & v_2 \mapsto bv_1 & (3) \\ & v_2 \mapsto a & (4) \end{array}$$

2. Draw a finite-state machine which accepts those strings with digits 0, 1, and 2 with the property that the sum of the digits in the string is a multiple of 3. (Note: The empty string Λ is trivially accepted by this machine.)

3. Write a regular expression for the language accepted by this grammar.

   $G = (V, S, v_0, \mapsto)$

   $V = \{v_0, v_1, v_2, 0, 1\}, \quad S = \{0, 1\}$

   $$\begin{array}{llr} \mapsto: & v_0 \mapsto 11v_1 & (1) \\ & v_1 \mapsto 1v_0 & (2) \\ & v_1 \mapsto 00v_2 & (3) \\ & v_2 \mapsto 00v_2 & (4) \\ & v_2 \mapsto 0 & (5) \end{array}$$

4. Consider the following language, consisting of just eight strings:

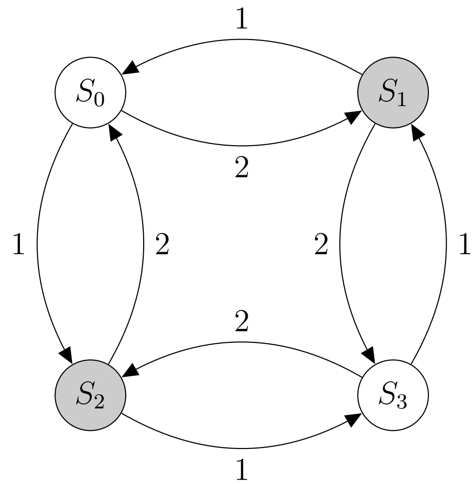
   $$11111 \vee 1112 \vee 1121 \vee 1211 \vee 2111 \vee 122 \vee 212 \vee 221.$$

   This may be shortened by factoring, as in:

   $$111(11 \vee 2) \vee 1121 \vee 1211 \vee 2111 \vee 122 \vee 212 \vee 221.$$

   Shorten this regular expression as much as possible. In this problem, one regular expression is *shorter* than another if it contains fewer digits (such as 1 or 2).

5. For the following finite-state machine, describe the language accepted by the machine in a brief sentence, and write a regular expression for the language. Here, S_0 is the start state, and accepting states are shaded in gray.



6. $G = (V, S, v_0, \mapsto)$

$V = \{v_0, v_1, a, b, \}, \quad S = \{a, b\}$

$$\begin{array}{lll} \mapsto: & v_0 \mapsto av_1 & (1) \\ & v_0 \mapsto bv_1 & (2) \\ & bv_1 \mapsto bav_1 & (3) \\ & bv_1 \mapsto bbv_1 & (4) \\ & av_1 \mapsto abv_1 & (5) \\ & v_1 \mapsto \Lambda & (6) \end{array}$$

Here, recall that Λ represents the empty string.

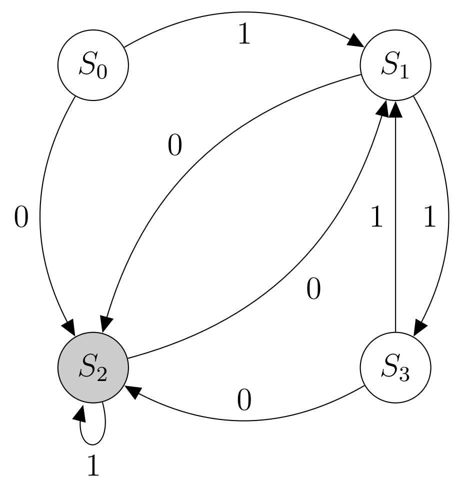
For this grammar,

(a) Show how to derive the string *abba*.

(b) Show how to derive the string *babbaba*.

(c) Describe the strings accepted by this grammar in a sentence or two, and write a regular expression for the language described by this grammar.

7. For the following finite-state machine, describe the language accepted by the machine in a brief sentence, and write a regular expression for the language. S_0 is the starting state, and S_2 is the accepting state.



8. Draw a finite-state machine that accepts all strings with letters a, b, and c such that abc occurs nowhere in the string.

9. $G = (V, S, v_0, \mapsto)$

   $V = \{v_0, v_1, a, b, c\}, \quad S = \{a, b, c\}$

   $$\mapsto: \begin{array}{ll} v_0 \mapsto 11v_0 & (1) \\ v_0 \mapsto v_1 & (2) \\ v_1 \mapsto v_1 000 & (3) \\ 111v_1 00 \mapsto v_1 & (4) \\ v_1 \mapsto 10 & (5) \end{array}$$

   For this grammar,

   (a) Show how to derive the string 10.

   (b) Show how to derive the string 100.

   (c) Write a regular expression for the language described by this grammar.

10. Decide which strings are accepted by the following grammar. Write a regular expression describing those strings.

    $G = (V, S, v_0, \mapsto)$

    $V = \{v_0, 0, 1\}, \quad S = \{0, 1\}$

    $$\mapsto: \begin{array}{ll} v_0 \mapsto v_0 00 & (1) \\ v_0 \mapsto 111v_0 & (2) \\ v_0 000 \mapsto 11v_0 & (3) \\ 1111v_0 \mapsto v_0 0 & (4) \\ v_0 \mapsto 1 & (5) \end{array}$$