CS320: Problems for Day 5, Winter 2023

Problem 1 Let L be the language over alphabet $\{a, b, c\}$ consisting of all strings that do not contain ab as a substring.

(a) Construct a finite automaton M that accepts L. If such an automaton M does not exist, explain why.

(b) If you constructed an automaton M in your answer to part (a), is M deterministic? Justify briefly your answer.

Problem 2 Let L be the language defined by the regular expression

$$\boldsymbol{a^*(ab \cup ba \cup e)b^*}$$

(a) Construct a finite automaton M that accepts L.

(b) Is the finite automaton M that you constructed in your answer to part (a) deterministic? Justify briefly your answer.

Problem 3 Let M be the finite automaton represented by the state diagram on Figure 1, and let L be the language accepted by M.

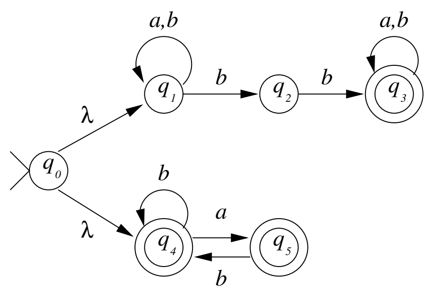

Figure 1:

(a) Write 10 distinct strings that belong to L.

(b) Write 10 distinct strings over alphabet $\{a, b\}$ that do not belong to L.

Problem 4 Let M be the finite automaton represented by the state diagram on Figure 2, and let L be the language accepted by M.

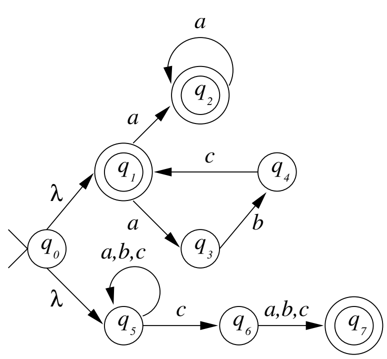

Figure 2:

(a) Write 10 distinct strings that belong to L.

(b) Write 10 distinct strings over alphabet $\{a, b, c\}$ that do not belong to L.

Problem 5 Let L be the language defined by the regular expression:

$$\boldsymbol{(a \cup bc)^*(dd \cup g)^* \cup ab^*}$$

Construct a state-transition graph of a finite automaton M that accepts L. If such automaton does not exist, prove it.

Problem 6 Let L be the language defined by the regular expression:

$$(\boldsymbol{a} \cup \boldsymbol{c})^* \, (\boldsymbol{b} \cup \boldsymbol{c})^* \cup \boldsymbol{bbb}$$

(a) Construct a state-transition graph of a finite automaton that accepts L. If such automaton does not exist, prove it.

(b) Does there exist an algorithm to convert an arbitrary regular expression into an equivalent finite automaton? Explain your answer briefly.

Problem 7 Let L be the set of strings over alphabet $\{a, b, c\}$ that do not contain the substring aa.

(a) Draw a state-transition graph of a finite automaton that accepts L. If such automaton does not exist, prove it.

(b) Is the complement $\overline{L}$ of the language L countable? Explain your answer briefly.

Problem 8 Let L be the set of strings over alphabet $\{a, b, c\}$ that do not start with c and do not end with a.

(a) Draw a state-transition graph of a finite automaton that accepts $\overline{L}$ (the complement of L). If such automaton does not exist, prove it.

(b) Does there exist an algorithm that solves the following problem:

INPUT: An arbitrary regular expression e.

OUTPUT: A finite automaton M_1 that accepts the complement of the language defined by e.

Explain your answer briefly.

Problem 9 Let L be the set of all strings over alphabet $\{a, b\}$ in which all a's come before all b's, and the number of a's is odd but the number of b's is even.

(a) Draw a state-transition graph of a finite automaton that accepts L. If such automaton does not exist, prove it.

(b) Write a complete formal definition of a context-free grammar that generates L. If such grammar does not exist, prove it.

Problem 10 Let L be the set of strings over alphabet $\{a, b, c\}$ whose total number of a's and b's is at least 2.

(a) Write a regular expression that defines L. If such a regular expression does not exist, explain why.

(b) Construct a state transition graph of a finite automaton that accepts L. If such an automaton does not exist, explain why.

(c) Write a complete formal definition of a context-free grammar G that generates L. If such a grammar does not exist, explain why.

Problem 11 Let L be the set of all nonempty strings over alphabet $\{a, b, d\}$ whose first symbol is equal to the third symbol.

(a) Write a regular expression that defines L. If such a regular expression does not exist, prove it.

(b) Construct a state-transition graph of a finite automaton that accepts L. If such an automaton does not exist, prove it.