TDDD85
Formal Languages and Automata Theory
2019-06-04

Materials allowed (Tillåtna hjälpmedel):

- A sheet of notes - 2-sided A5 or 1-sided A4. These notes must be handed in together with the answers and signed in the same way as the exam papers. (Ett blad med anteckningar - 2-sidigt A5 eller 1-sidigt A4. Detta blad ska lämnas in med svaren och signeras på samma sätt som övriga papper.)
- An english dictionary. (Engelsk ordbok).

Instructions:

- You may answer in english or swedish.
- Make sure your text and figures are big and clear enough to read easily.
- All answers must be motivated. A correct answer without reasonable motivation may result in zero points!

Grading: The maximum number of points is 34. The grades are as follows:

grade	TDDD85
3:	15–21 p.
4:	22–27 p.
5:	28–34 p.

Problems

1. Assume the alphabet $\Sigma = \{0, 1, 2\}$. Define the sum $sum(s)$ of a string $s \in \Sigma^*$ such that (6 p)

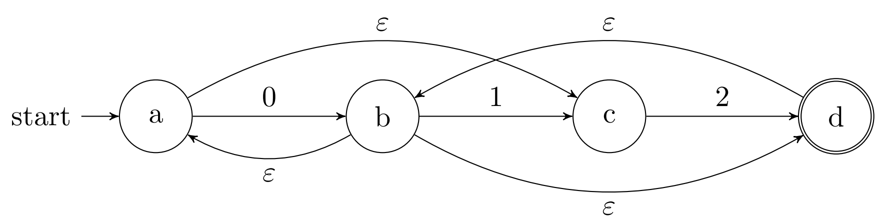
   - $sum(\varepsilon) = 0$ and
   - if $s = x_1, \ldots, x_n$ for some $n > 0$, then $sum(s) = \sum_{i=1}^{n} x_i$,

   i.e. the sum of s is the sum of all its symbols. For example, $sum(0121) = 0 + 1 + 2 + 1 = 4$ and $sum(101120) = 1 + 0 + 1 + 1 + 2 + 0 = 5$.

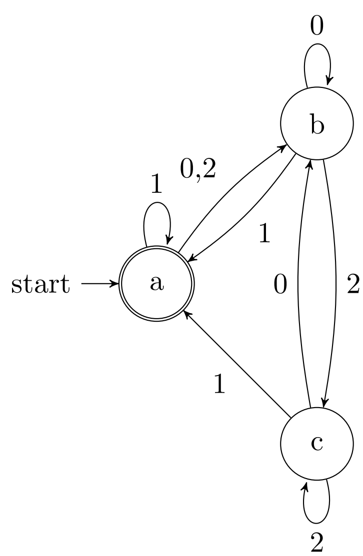
   (a) Draw the state diagram for a DFA that accepts the language $L_1 = \{s \in \Sigma^* \mid sum(s) \text{ is odd }\}$. For instance, 010, 102 and 1121 are in L_1, but 0, 112 and 2011 are not in L_1.

(b) Draw the state diagram for a DFA that accepts the language $L_2 = \{s \in \Sigma^* \mid sum(s) = 3k \text{ for some integer } k \geq 0\}$, i.e. all strings s such that $sum(s)$ modulo 3 is 0. For instance, 0, 111 and 10221 are in L_2, but 11, 0202 and 11011 are not in L_2.

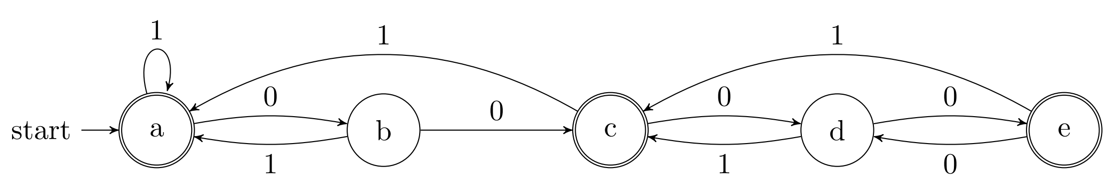
2. Convert the following NFA to a DFA using the subset construction method. Also draw the state diagram for the resulting DFA. (4 p)



3. Convert the following DFA to a regular expression using the GNFA method (or one of the other standard methods in the course). (4 p)



4. Show that the following DFA has a minimal number of states or construct an equivalent DFA with a minimal number of states. Use the algorithm from the course and specify clearly what is marked in each stage of the algorithm. Also draw the state diagram of the resulting minimal DFA. (4 p)



5. (6 p)

(a) Prove that the language $L_1 = \{a^k b^m c^n \mid 0 < k \leq n, m > n - k\}$ is not regular by using the pumping lemma for regular languages.

(b) Prove that the language $L_2 = \{a^{3k} b^m c^k b^n \mid 0 < k, 0 < n < m\}$ is not context free by using the pumping lemma for context-free languages.

6. Consider the following context-free grammar: (4 p)

$$
\begin{array}{llll}
G: & S & \rightarrow & A \mid bB \\
 & A & \rightarrow & aA \mid \varepsilon \\
 & B & \rightarrow & bB \mid Ab \mid \varepsilon
\end{array}
$$

(a) Show that G is ambiguous.

(b) Convert G to Chomsky normal form.

7. Let L_1 and L_2 be two languages. Define $L = L_1 \setminus L_2$, where "$\setminus$" denotes set difference, i.e. $A \setminus B = A \cap \overline{B}$. (6 p)

(a) Prove that L is Turing recognizable if L_1 is Turing recognizable and L_2 is decidable.

(b) Prove or explain why L may not be Turing recognizable if L_1 is decidable and L_2 Turing recognizable.

Note that the terminology differes between books and that
Turing recognizable = semi-decidable = recursively enumerable and
decidable = recursive.