Theoretical Computer Science

Winter semester 21/22

Prof. Dr. Georg Schied

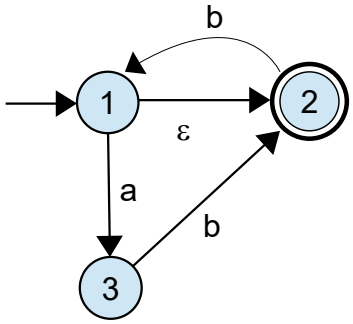
Assignment 7

Deadline: Wednesday, 24 November 2021

10 out of 20 points have to be achieved in order to pass.

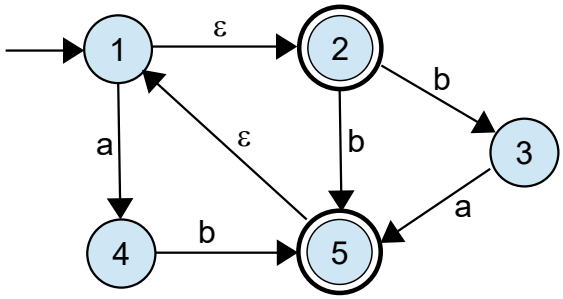
Exercise 7.1

Convert the following ε-NFA into a DFA using the Rabin-Scott subset construction (Definition 8.13).



Exercise 7.2

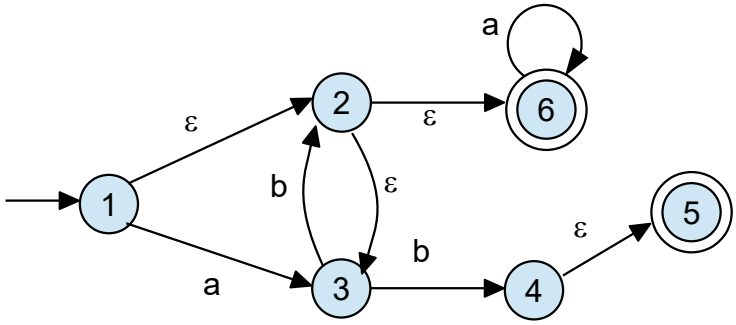
The following ε-NEA with alphabet $\Sigma = \{a, b\}$ is given:



Convert the automaton to an equivalent deterministic finite automaton (DFA) using the improved subset construction (Algorithm 8.19).

Exercise 7.3 - obligatory (7 points)

Let the following ε-NFA be given:



a) If you would use the Rabin-Scott *subset construction* from Definition 8.13 to convert this ε-NFA into an DFA, how many states would the resulting DFA have?

b) Use the improved subset construction (Algorithm 8.19) to convert this ε-NFA into an equivalent DFA.

Exercise 7.4 - obligatory (6 points)

A set S of specific arithmetic expressions is inductively defined in the following way:

(1) The numbers 6, 15 and 33 are contained in S.

(2) If expressions s_1 and s_2 are contained in S, then also the following expressions are contained in S:

$(s_1 + s_2)$

$(s_1 \cdot s_2)$

s_1^2

An example of such an expression is $((33 + (15 \cdot 6)^2) \cdot 33^2)$.

Prove by *structural induction* that evaluation of each expression $s \in S$ produces a value that is divisible by 3.

Exercise 7.5

Systematically build an ε-NFA (using procedure 9.1) that accepts the language of the following regular expression:

a*(ba|c)*

Exercise 7.6 - obligatory (7 points)

Let the following regular expression be given:

b(ca)* | aa

a) Describe the syntactic structure of the regular expression as an abstract syntax tree.

b) Use the inductive method presented in procedure 9.1 to build an ε-NFA that accepts the language of regular expression.