Homework 3
Automata

Version du September 20, 2020

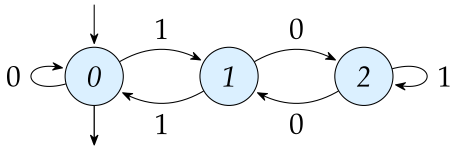
This homework has to be returned tomorrow, on Thursday, at the beginning of the tutorial.

Exercise 1 (Translation of rational/regular expressions). *In this exercise, we assume that we have $\Sigma = \{a, b, c\}$.*

For each following rational expression, you have to use the Thompson algorithm to build a finite nondeterministic automaton with spontaneous transitions, and then apply the building procedure seen during the lessons to get rid of spontaneous transitions, and finally the potentialy unuseful states[1]*. You will show the automata after each step.*

1. $(a + b + cc)^\star abab$
2. $((ab + \varepsilon)^\star c)^\star$
3. $(\emptyset(a + b))^\star$

Exercise 2 (Multiples of 3 and 7). *Let $\mathcal{D}_3$ be the following deterministic automaton defined on the alphabet $\Sigma = \{0, 1\}$:*



1. *Execute $\mathcal{D}_3$ on the words* 101010*, and* 11111*.*
2. *Prove that $\mathcal{D}_3$ recognizes the binary representations of the narutal integers multiple of* 3*. (Hint: give a meaning to the states numbers.)*
3. *Build $\mathcal{D}_7$, a deterministic automaton which recognizes the binary representations of the natural numbers which are multiple of* 7*.*

Exercise 3 (Automaton for entry code). *You will have understood that the program of the entry code of yesterday (Howework 2, exercise 1) only executed an automaton. Today, we are going to modify the table* `tab` *of the program of Homework 2 to recognize another code.*

1. *Does the table* `tab` *of the preceding homework represent an NFA? a DFA?*
2. *How is the initial state indicated? And how about the final states?*
3. *We want to change the code in such a manner that the entry code only accepts the sequences of numbers which end by* `747`*. Give a rational expression denoting the set L of these sequences.*
4. *Using the Thompson algorithm, build an ε-NFA which recognizes L. Then, delete the spontaneous transitions to obtain an NFA. Prune if necessary. Salt and pepper.*
5. *Build a DFA able to recognize L. You can proceed as you want; you are not been told how to. (Hint: you need only 4 states.)*
6. *Give a new implementation of the program of the entry code, able to recognize the words of L (followed by "*`E`*", as usual) based on the same principle as in Homework 2.*

[1] see the section about *Unuseful states* in the course