Talen en Automaten

Assignment 3, Tue 28th Nov, 2017

Exercise teachers. The student groups are supervised by the following teachers:

Teacher	E-Mail	Room	Time
Michiel de Bondt	M.deBondt@math.ru.nl		8:45 – 10:30
Demian Janssen	wd.janssen@student.ru.nl		8:45 – 10:30
Leon Gondelman	lgondelmann@gmail.com		8:45 – 10:30
Tom van Bussel	tom.van.bussel@student.ru.nl		8:45 – 10:30
David Venhoek	david@venhoek.nl		8:45 – 10:30
Alexis Linard	A.linard@cs.ru.nl		8:45 – 10:30
Bas Steeg	bas.steeg@student.ru.nl		10:45 – 12:30
Ties Robroek	ties.robroek@student.ru.nl		10:45 – 12:30
Jan Martens	j.martens@student.ru.nl		15:45 – 17:30

Postboxes are located in the Mercator building on the ground floor There will be boxes labelled with *Talen en Automaten* and the corresponding group teacher's name. There will be 1 box, the *Uitleverbak*, for work that hasn't been picked up at the exercise hours.

Handing in your answers: There are two options:

1. E-mail: Send your solutions by e-mail to your exercise class teacher (see above) with subject "**T&A: assignment 3**". This e-mail should only contain a single PDF document as attachment (unless explicitly stated otherwise). Before sending an e-mail make sure:
   - the file is a PDF document
   - your name is part of the filename (for example MyName_assignment-3.pdf)
   - your name and student number are included in the document (they will be printed).
2. Post box: Put your solutions in the appropriate post box (see above). Before putting your solutions in the post box make sure:
   - your name, student number, and IC, KI or Wiskunde are written clearly on the document.

Deadline: Tue 5th Dec, 2017, 8:45 (in Nijmegen!)

Goals: After completing these exercises successfully you should be able to construct an NFA from a language description, to construct an NFA-λ from a regular expression, to turn an NFA-λ into an NFA and to determinise an NFA.

There are 3 mandatory exercises, worth **10 points** in total. There is 1 more, extra hard, exercise. Be aware that this exercise is just for fun, you cannot earn any points with it.

1 NFAs and Their Languages

Let $A = \{0, 1, 2\}$ and let L be the set of words in which the last digit occurs at least twice, with no larger digit in between the last two occurences:

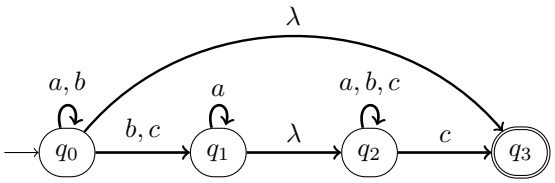
$$L = \{w \in A^* \mid \exists x \in A.\ \exists u, v \in A^*.\ w = uxvx, \text{ and there is no } y \text{ in } v \text{ s.t. } x < y\},$$

where $0 < 1 < 2$. For example, $00, 2111, 2102 \in L$, but $1, 121 \notin L$.

Construct an NFA that accepts L, and show that 2101 is accepted, but 121 not. **(4pt)**

2 From NFA-λ to NFA

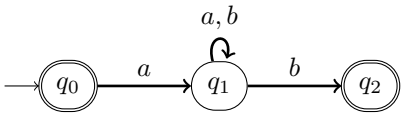
Let M be the NFA-λ over the alphabet $A = \{a, b, c\}$ given by the following graph.



Give an NFA (without λ-transitions) with the same set of states, accepting the same language. **(2pt)**

3 From NFA to DFA

Let M be the NFA over the alphabet $A = \{a, b\}$ given by the following graph.



Use the powerset construction from the lecture to turn M into a DFA M' that accepts the same language. Leave out unreachable states, and clearly indicate how a state of M' corresponds to a subset of states of M. **(2pt)**

4 From Regular Expressions to NFAs-λ

Let e be the regular expression $(b + a^*)a$.

Use the "toolkit" from the lecture to construct an NFA-λ that accepts $\mathcal{L}(e)$. The (non-trivial) intermediate steps must be given as part of the solution. **(2pt)**

5 Fun Exercises – Properties of Regular Languages

a) Using that regular languages are closed under complement and intersection, show that for regular languages L_1, L_2 their difference $L_1 \setminus L_2$ is regular as well.

b) Give an algorithm that decides whether for a regular expression e its language $\mathcal{L}(e)$ is empty.

c) Give an algorithm that checks for given regular expressions e_1, e_2 whether their languages are equal: $\mathcal{L}(e_1) = \mathcal{L}(e_2)$.

6 Fun Exercise – Constructing an NFA-λ

Give an NFA-λ over $A = \{a\}$ such that it rejects some string and the length of the shortest rejected string is strictly greater than the number of states.