Talen en Automata

Assignment 6, Tuesday 9th January, 2018

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

Teacher	E-Mail	Room	Time
Michiel de Bondt	`M.deBondt@math.ru.nl`	HG00.310	8:45 – 10:30
Demian Janssen	`wd.janssen@student.ru.nl`	HG01.058	8:45 – 10:30
Leon Gondelman	`lgondelmann@gmail.com`	HG00.065	8:45 – 10:30
Tom van Bussel	`tom.van.bussel@student.ru.nl`	HG02.028	8:45 – 10:30
David Venhoek	`david@venhoek.nl`	HG03.632	8:45 – 10:30
Alexis Linard	`A.linard@cs.ru.nl`	HG00.058	8:45 – 10:30
Bas Steeg	`bas.steeg@student.ru.nl`	HG00.062	10:45 – 12:30
Ties Robroek	`ties.robroek@student.ru.nl`	HG00.062	10:45 – 12:30
Jan Martens	`j.martens@student.ru.nl`	HG00.308	15:45 – 17:30
Nienke Wessel	`N.Wessel@student.ru.nl`	HG00.310	15:45 – 17:30

Postboxes are located in the Mercator building on the ground floor. There are 7 boxes labelled with *Talen en Automaten* and the corresponding group teacher's name. There is 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 6**". 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-6.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: Tuesday 16th January, 2018, 13:45 (in Nijmegen!)

Goals: After completing these exercises successfully you should be able to show that a language is context free by giving a push down automaton (PDA) that accepts it. Moreover, you should be able to turn a context free grammar into a PDA and vice versa.

There are 2 mandatory exercises, worth **10 points** in total. There are 2 more, extra hard, exercises. Be aware that these exercises are just for fun, you cannot earn any points with them.

1 Push Down Automata

a) Let L_1 be given by

$$L_1 = \{wcv \in \{a,b,c\}^* \mid w,v \in \{a,b\}^* \text{ and } |v|_a = 2|w|_a\}$$

Construct a *deterministic* PDA that accepts L_1. Explain your answer. **(2pt)**

b) Let L_2 be given by

$$L_2 = \{w \in \{a,b\}^* \mid |w|_a = |w|_b\}.$$

Construct a PDA that accepts L_2, explain your answer (!), and show that the word *abba* is accepted but *aa* is not. **(3pt)**

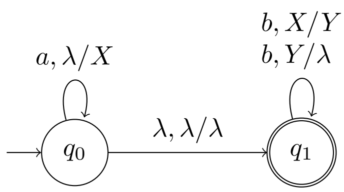
2 CFGs and PDAs

a) The following grammar generates expressions such as $(a+a)+a$.

$$S \longrightarrow F \mid S+S$$
$$F \longrightarrow a \mid (S)$$

Use the construction given in the lecture to give a two state PDA that accepts the language generated by this grammar. **(2pt)**

b) Let M be the PDA over the alphabet $\{a,b\}$ and with stack alphabet $\{X,Y\}$ given by the following graph.

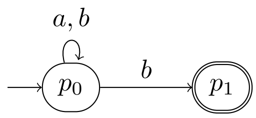


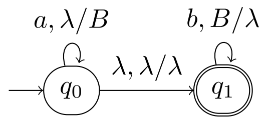
Use the procedure given in the lecture to construct a grammar that generates $\mathcal{L}(M)$, and give the derivation for the word *abb*. **(3pt)**

3 Fun Exercises – Closure Properties of CFLs

a) Let $\mathcal{A} = (Q_1, q_{1,0}, F_1, \delta_1 : Q_1 \times \Sigma \to \mathcal{P}(Q_1))$ be an NFA accepting the regular language L and let $\mathcal{B} = (Q_2, \Sigma, \Gamma, q_{2,0}, F_2, \delta_2 : Q_2 \times \Sigma_\lambda \times \Gamma_\lambda \to \mathcal{P}(Q_2 \times \Gamma_\lambda))$ be a PDA that accepts the context free language D. Show that $L \cap D$ is context free by defining a PDA accepting it. Use $Q_1 \times Q_2$ as state space.

b) Let $\mathcal{A}$ and $\mathcal{B}$ be given as in the following diagrams.





Apply your construction from **3a)** to these automata to define a PDA that accepts $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B})$.

4 Fun Exercises – Beyond CFLs

We extend PDAs to two-stack PDAs (PDA_2). A PDA_2 M is given by a tuple $(Q, \Sigma, \Gamma, \delta, q_0, F)$ just like a PDA, except that

$$\delta : Q \times \Sigma_\lambda \times \Gamma_\lambda \times \Gamma_\lambda \to \mathcal{P}(Q \times \Gamma_\lambda \times \Gamma_\lambda).$$

Give a suitable extension of the acceptance condition for PDAs, and show that two-stack PDAs are computationally more powerful than PDAs, by showing that there is a PDA_2 that accepts

$$\{a^n b^n c^n \mid n \in \mathbb{N}\}.$$