Department of Information and Computing Sciences
Utrecht University

INFOB3TC – Exam 2

Johan Jeuring

Monday, 25 January 2016, 8:30–10:30

Preliminaries

- The exam consists of 8 pages (including this page). Please verify that you got all the pages.
- Fill out the answers **ON THE EXAM ITSELF**.
- Write your **name** and **student number** here:

- The maximum score is stated at the top of each question. The total amount of points you can get is 90.
- Try to give simple and concise answers. Write readable text. Do not use pencils or pens with red ink. You may use Dutch or English.
- When writing grammar and language constructs, you may use any set, sequence, or language operations covered in the lecture notes.
- When writing Haskell code, you may use Prelude functions and functions from the following modules: *Data.Char*, *Data.List*, *Data.Maybe*, and *Control.Monad*. Also, you may use all the parser combinators from the `uu-tc` package. If you are in doubt whether a certain function is allowed, please ask.

Good luck!

Questions

Regular expressions, languages and pumping lemmas

1 (10 points)**.** Consider the DFA (X, Q, d, S, F) where $X = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$, $Q = \{q_1, q_2\}$, d is defined by:

$d\ q_1\ \mathtt{a} = q_1$
$d\ q_1\ \mathtt{b} = q_2$
$d\ q_2\ \mathtt{a} = q_1$
$d\ q_2\ \mathtt{c} = q_2$

$S = q_1$, and $F = \{q_2\}$. Give the regular expression denoting the language accepted by this automaton. •

For the following three tasks: consider the following three languages:

$$\begin{aligned} L_1 &= \{a^n b^m c d^m e^n | n, m \geqslant 0\} \\ L_2 &= \{(ab)^n c d^m | n, m \geqslant 0\} \\ L_3 &= \{a^n b (cd)^n e^n | n \geqslant 0\} \end{aligned}$$

2 (5 points)**.** One of the languages is regular, one context-free and not regular and one not context-free. Which are the regular and the non-regular context-free languages? •

3 (5 points)**.** Give a regular grammar for the regular language, and a context-free for the context-free language. •

4 (10 points)**.** Prove that the grammar that is context-free but not regular is indeed not regular by using the pumping lemma for regular languages . •

LR parsing

Consider the following grammar:

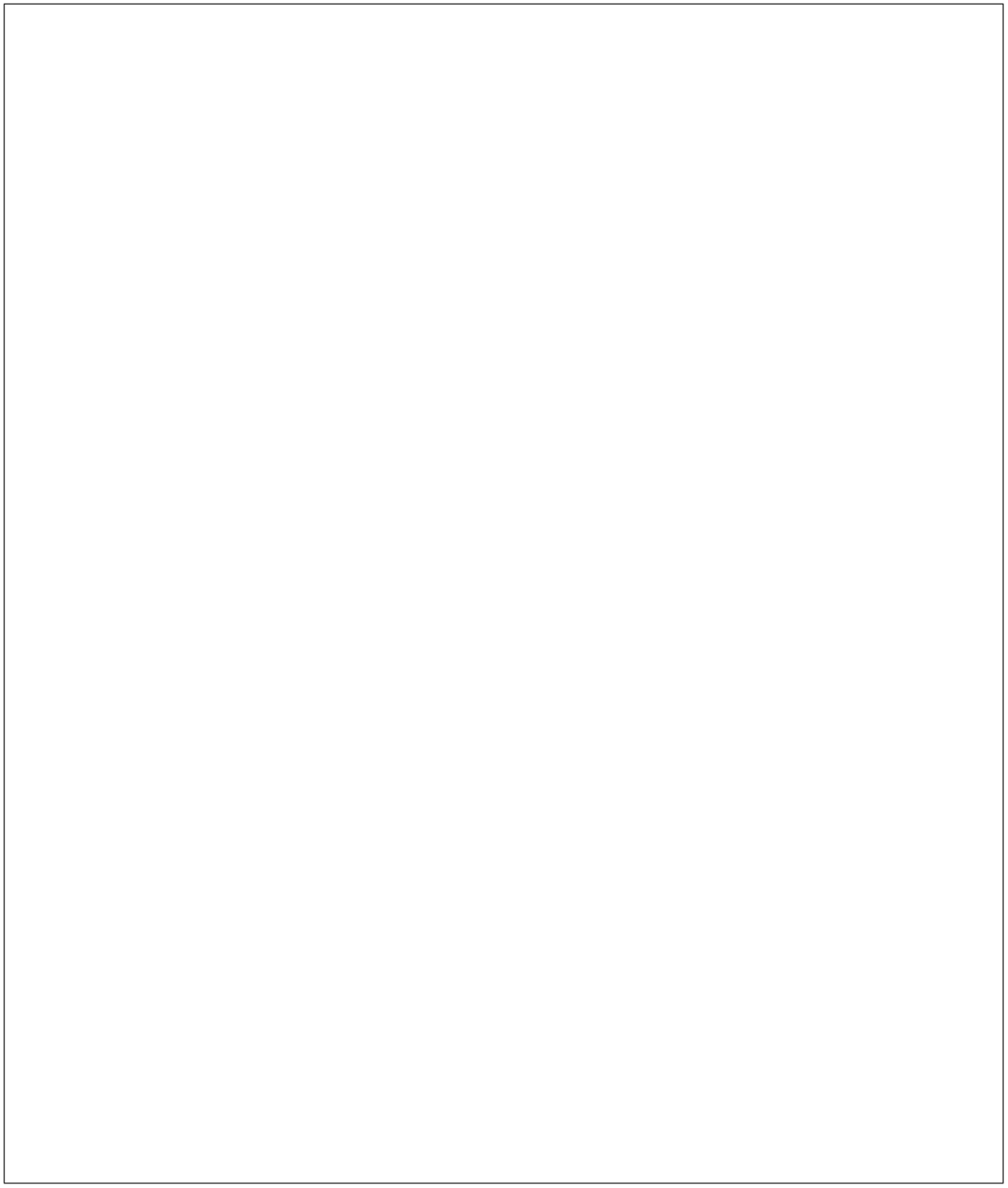
$S \rightarrow ABC\$$
$A \rightarrow \texttt{a}$
$A \rightarrow \texttt{a}C$
$B \rightarrow \texttt{b}$
$B \rightarrow \texttt{b}C$
$C \rightarrow \texttt{c}$

5 (10 points)**.** This grammar is not LR(0). Construct the LR(0) automaton for this grammar, and show which conflicts appear where. •

6 (10 points)**.** Is this grammar SLR(1)? If so, construct the SLR-table. If not, explain where you cannot make a choice in a shift/reduce conflict or a reduce/reduce conflict.

•

7 (10 points)**.** Play through the LR parsing process for the sentence "`acbcc$`". If there is a choice somewhere, make this explicit. Show in each step at which state in your LR(0) automaton you are. •



LL parsing

In these exercises we will look at the grammar

$S \to AB$
$A \to \mathtt{a}A\mathtt{a} \mid \varepsilon$
$B \to \mathtt{b}B\mathtt{b} \mid \varepsilon$

8 (15 points)**.** Complete the table below by computing the values in the columns for the appropriate rows. Use *True* and *False* for property values and set notation for everything else.

NT	Production	*empty*	*emptyRhs*	*first*	*firstRhs*	*follow*	*lookAhead*
S							
	$A \to AB$						
A							
	$A \to \mathtt{a}A\mathtt{a}$						
	$A \to \varepsilon$						
B							
	$B \to \mathtt{b}B\mathtt{b}$						
	$B \to \varepsilon$						

•

9 (10 points)**.** Is the above grammar LL(1)? Explain how you arrived at your answer. If the grammar is not LL(1), give a grammar that generates the same language and is LL(1). •

10 (5 points)**.** Show the steps that a parser for the above LL(1) grammar goes through to recognize the following input sequence:

```
aabb
```

For each step (one per line), show the stack, the remaining input, and the action (followed by the relevant symbol or production) performed. If you reach a step in which you cannot proceed, note the action as "error." •