EXAM REPORT

Advanced Programming 2013

Rasmus Borgsmidt

CONTENTS

ACRONYMS

INTRODUCTION

1 SALSA LANGUAGE PARSER

1.1 *Choice of Parser Library*

1.2 *Grammar Transformations*

Assumptions: || is left-associative.
'+' and '-' have the same precedence for expressions
As outlined by Mogensen [1, p. 69].

1. Eliminate ambiguity
2. Eliminate left-recursion
3. Perform left-factorization where required

Note: And it is only required when several productions *for the same nonterminal* begin with the same sequence of symbols; for example, it is not a problem that both *Pos* and *Prim* have a production that begins with '(', as long as they have only one each.

The resulting grammar makes it explicit that '@' has higher precedence than '||'. In other words, it makes it impossible to derive the wrong parse tree from a given input.

2 SALSA LANGUAGE INTERPRETER

3 ATOMIC TRANSACTION SERVER

4 CONCLUSION

BIBLIOGRAPHY

[1] Torben Æ. Mogensen. *Introduction to Compiler Design*. Springer-Verlag London Ltd., 2011.

A SALSA GRAMMAR

B LEXICAL SPECIFICATION

C SUBMITTED FILES

Sample programs; example **??** shows how to compile and run a sample program.

Listing 1: File tree under `src/`

```
.
|-- examples
|   |-- id.class
|   |-- id.ial
|   |-- id.java
|   |-- id2.class
|   |-- id2.ial
|   |-- id2.java
|   |-- optimized.ial
|   |-- optimized.java
|   |-- plus.class
|   |-- plus.ial
```

Table 1: Syntax of target language

Program	::=	*DefComs*
DefComs	::=	*DefCom DefComs*$_*$
DefComs$_*$	::=	*DefCom DefComs*$_*$
	|	ϵ
DefCom	::=	*Command*
	|	*Definition*
Definition	::=	`'viewdef'` *VIdent Expr Expr*
	|	`'rectangle'` *SIdent Expr Expr Expr Expr Colour*
	|	`'circle'` *SIdent Expr Expr Expr Colour*
	|	`'view'` *VIdent*
	|	`'group'` *VIdent* `'['` *VIdents* `']'`
Command	::=	*Command1 Command*$_*$
Command$_*$	::=	`'||'` *Command1 Command*$_*$
	|	ϵ
Command1	::=	*Command2 Command1*$_*$
Command1$_*$	::=	`'@'` *VIdent Command1*$_*$
	|	ϵ
Command2	::=	*SIdents* `'->'` *Pos*
	|	`'{'` *Command* `'}'`
VIdents	::=	*VIdent VIdents*$_*$
VIdents$_*$	::=	*VIdent VIdents*$_*$
	|	ϵ
SIdents	::=	*SIdent SIdents*$_*$
SIdents$_*$	::=	*SIdent SIdents*$_*$
	|	ϵ
Pos	::=	`'('` *Expr* `','` *Expr* `')'`
	|	`'+' '('` *Expr* `','` *Expr* `')'`
Expr	::=	*Prim Expr*$_*$
Expr$_*$	::=	`'+'` *Prim Expr*$_*$
	|	`'-'` *Prim Expr*$_*$
	|	ϵ
Prim	::=	*integer*
	|	*SIdent* `'.'` *Coord*
	|	`'('` *Expr* `')'`
Coord	::=	`'x'` | `'y'`
Colour	::=	`'blue'` | `'plum'` | `'red'` | `'green'` | `'orange'`