

Introduction to compilers - G-milestone

Sander Tams
KU-id: qlm300
DOB: 021091

Arinbjörn Brandsson
KU-id: hkt789
DOB: 280292

Julie Engholm
KU-id: hwx240
DOB: 051089

December 10, 2013

Contents

1	Project tasks	1
1.1	A Parser for Paladim Using the mosmlyac Parser Generator	1
1.2	Integer Multiplication and Division, and Boolean Operators	1
1.3	Simple Type Inference and Type Checking the new Function	1
1.4	Type Checking and Code Generation for Array Indexing	1
1.5	Call-by-Value-Result Semantics for Procedures	1
A	Code	1

1 Project tasks

1.1 A Parser for Paladim Using the mosmlyac Parser Generator

Our parser for the Paladim programming language is implemented. While our implementation works for the specified grammar, we had to change some of the grammar slightly in order to prevent a shift/reduce conflict. A consequence of this would be that, instead of writing

```
1 If <expr> Then <Statement>;
```

We must write

```
If <expr> Then begin <Statement>; end;
```

else the parser won't parse the program correctly. For our final group assignment, we will have this fixed.

1.2 Integer Multiplication and Division, and Boolean Operators

Will be included in milestone

1.3 Simple Type Inference and Type Checking the new Function

Will be included in final submission

1.4 Type Checking and Code Generation for Array Indexing

Will be included in final submission

1.5 Call-by-Value-Result Semantics for Procedures

Will be included in final submission

A Code

This is the appendix where all our code, and relevant snippets of the handed-out implementation, goes.