Bachelor Project in Compiler Construction
February 2019

Report from group GROUPNUMBER: 9

Anton Nørgaard (antno16), Bjørn Glue Hansen (bhans09) & Thor Skjold Haagensen (thhaa16)

1 Introduction

1.1 Implementation Status

1.2 Scope Rules

1.3 Symbol Data

2 Typecheck

2.1 Types

Weeding and Typechecking

It is a policy of the compiler that we will allow a file to compile if it lacks sufficient return statements, but emit a warning if so. The general principle for checking if a function returns is that it has a list of statements and if at least one of them has a valid return statement, the function validly returns

When diciding the validity of a function's return statement three cases occur. The simplest is that the function has a basic return statement, in this case varifying is simple.
If the return statement lies in an if then else statement then both the if and the else part must have a return statement or the return statement must be outside of the if and else.
Finally if the statement we check is a list of statements, a return statement must be found in this list.
In order to ensure that a function is defined and ends with the same identifier name,

Additions to the symbol ta-

ble A few of the significant changes we've made to the symbol table is we have added more information to each symbol in the table. Namely, we've added a kind to what the symbol denotes, e.g whether it is a variable, function, type, etc. And included the values function and type, which we can use to validate whether a value is used correctly or if a function has a proper definition/use

2.2 Type Rules

2.3 Algorithm

For instance, the rule

2.4 Test

Below table shows the results of the tests. Op is an abbreviation for binary operators.

#	Test	Expected Result	Pass
	Parsing.sh: Boolean Precedence Tests		
1	Boolean ops are left most associative.	Inner parentheses around first op.	✓
2	&& op has higher precedence than || op.	Inner parentheses around && op.	✓
	Parsing.sh: Comparison Association Tests		