

# MiniJava Compiler (`mjc`)

Daniel Månsson  
dmans@kth.se

Elvis Stansvik  
stansvik@kth.se

XX May 2014

## Abstract

Some abstract

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Compiler Design</b>	<b>1</b>
2.1	Lexical Analysis and Parsing . . . . .	1
2.2	Semantic Analysis . . . . .	1
2.3	Pass Foo . . . . .	2
<b>3</b>	<b>Using the Compiler</b>	<b>2</b>
3.1	Building . . . . .	2
3.2	Running . . . . .	2
3.3	Example . . . . .	2
<b>4</b>	<b>Future Improvements</b>	<b>2</b>

## 1 Introduction

Some introduction to the project. [1]

## 2 Compiler Design

Some info about the compiler design.

### 2.1 Lexical Analysis and Parsing

Some info about lexical analysis / parsing.

### 2.2 Semantic Analysis

Some info about semantic analysis.

## 2.3 Pass Foo

Some info about pass foo.

# 3 Using the Compiler

## 3.1 Building

Some info about building the compiler.

## 3.2 Running

Some info about running the compiler.

```
usage: mjc <infile> [options]
  -S          output assembler code
  -o <arg>    output file
  -p          print abstract syntax tree
  -g          print abstract syntax tree in GraphViz format
  -s          print symbol table
  -h          show help message
```

## 3.3 Example

An example of invoking the compiler.

# 4 Future Improvements

Some notes about future improvements.

# References

- [1] Torbjörn Granlund and Andrew W. Appel. *Context-free grammar for Mini-java variant*. URL: <http://www.csc.kth.se/utbildning/kth/kurser/DD2488/komp14/project/grammar14v1.pdf>.