

# Compilerbau

Sommersemester 2004

10. Übungsblatt

Abgabetermin: 5. Juli 2004, 12 Uhr

Aufgabe 1:

20 Punkte

Gegeben sei die Sprache MiniJava, deren Syntax durch die folgende Grammatik definiert wird:

```
program → decl* stmt*
decl    → type name (, name)* ;
type    → int
stmt    → ; | {decl* stmt*} | name = expr; | name = read (); | write (expr); |
         if (cond) stmt | if (cond) stmt else stmt | while (cond) stmt
expr    → number | name | (expr) | unop expr | expr binop expr
unop    → -
binop   → - | + | * | / | %
cond    → true | false | (cond) | expr comp expr |
         bunop cond | cond bbinop cond
comp    → == | != | <= | < | >= | >
bunop   → !
bbinop  → && | ||
number  → sign? digit digit*
sign    → + | -
digit   → 0 | ... | 9
name    → letter (letter|digit)*
letter  → $ | _ | a | ... | z | A | ... | Z
```

Schreibe einen Parser, der eine einheitliche Einrückung für MiniJava-Programme erzeugt. Die Namen der Ein- und Ausgabedateien sollen aus der Kommandozeile gelesen werden. Verwende CUP und JFLEX zur Generierung des Parsers. Ein Beispiel-Parser, der als Grundlage für diese Aufgabe verwendet werden kann, sowie Muster für die Eingabe und die erwartete Ausgabe sind auf der Vorlesungsseite zu finden (<http://www2.in.tum.de/lehre/vorlesungen/SS04/compilerbau.html>).

Sende per Email alle benötigten Dateien inklusive CUP- und JFLEX-Spezifikationen, sowie einem Skript zur vollständigen Kompilierung des Parsers (von den Spezifikationen bis hin zum ausführbaren Java-Programm).

Links zu Software und Dokumentation:

Parser-Generator CUP: <http://www.cs.princeton.edu/~appel/modern/java/CUP/>

Scanner-Generator JFLEX: <http://jflex.de/>