The

CKPMcc

Programming Language

Compiler Construction Course

Summer 2006

Team Members:

Clemens Krainer 9020112

Salzburg, 13 July 2006

Content

1. Introduction

The CKPMcc programming language is a subset of the C programming language, as standardised in [1].

2. EBNF of the Compiler

2.1 Formal Definition

CharSet = '!', '#' .. '0xFF'.

Identifier = [Letter | '_']{Letter | Digit | '_'}.

String = '"' { CharSet } '"'.

Constant = Number | '\'' CharSet '\'' | String.

Number = Digit { Digit }.

Letter = 'A' .. 'Z' | 'a' .. 'z'.

Digit = '0' .. '9'.

2.2 Expressions

Program = { Declaration }.

Declaration = StructDeclaration | SimpleDeclaration | FunctionDeclaration.

FunctionDeclaration = TypeName [Pointer] Identifier "(" NameList ")" (";" | Block).

SimpleDeclaration = MemberDeclaration.

StructDeclaration = "struct" Identifier "{" MemberDeclaration { MemberDeclaration } "}" ";".

MemberDeclaration = TypeName [Pointer] Identifier ["[" Number "]"] ";".

TypeName = "void" | "char" | "int" | "bool" | "struct" Identifier.

Pointer = "*" { "*" }.

NameList = TypeName [Pointer] Identifier { "," TypeName [Pointer] Identifier }.

Block = "{" { DataDeclaration } { Statement } "}".

Statement = Block
| "if" "(" AssignmentExpression ")" Statement ["else" Statement]
| "while" "(" AssignmentExpression ")" Statement
| "break" ";"
| "continue" ";"
| "return" [AssignmentExpression] ";"
| AssignmentExpression ";".

AssignmentExpression = { [Pointer] Identifier Selector "=" } [TypeCastExpression] LogicalOrExpression.

TypeCastExpression = "(" TypeName [Pointer] ")".

LogicalOrExpression = LogicalAndExpression { "||" LogicalAndExpression }.

```
LogicalAndExpression   = ConditionalExpression { "&&" ConditionalExpression }.

ConditionalExpression  = SimpleExpression [ ("==" | "!=" | "<" | "<=" | ">=" | ">") SimpleExpression ] .

SimpleExpression       = UnaryOperator Term { ( "+" | "-" | "|" ) Term }.

UnaryOperator          = "+" | "-" | "!" | "&" | "*" | "~".

Term                   = Factor { ( "*" | "/" | "%" | "&" | "<<" | ">>" ) Factor }.

Factor                 = "sizeof" "(" TypeName ")"
                           | Number
                           | Character
                           | String
                           | "(" AssignmentExpression ")"
                           | Identifier ( "(" [ Parameter ] ")" | Selector }.

Selector               = { "." Identifier | "->" Identifier | "[" LogicalOrExpression "]" }.

Parameter              = AssignmentExpression { "," AssignmentExpression }.
```

3. Compiler Keywords

Used compiler keywords:
`if`, `else`, `while`, `struct`, `sizeof`, `return`, `break`, `continue`, `char`, `int`, `void`.

Not used compiler keywords:
`for`, `short`, `long`, `float`, `double`, `signed`, `unsigned`, `auto`, `register`, `typedef`, `union`, `const`, `volatile`, `goto`, `switch`, `case`, `default`, `do`, `elsif`, `static`, `extern`, `enum`.

4. References

[1] American National Standards Institute (1989) *Programming Language C*. ANSI X3.159-1989