
ALBERT LUDWIGS UNIVERSITÄT FREIBURG

TECHNISCHE FAKULTÄT

PicoC-Compiler

Übersetzung einer Untermenge von C in den Befehlssatz der RETI-CPU

BACHELORARBEIT

Abgabedatum: 28th April 2022

Author:
Jürgen Mattheis

Gutachter:
Prof. Dr. Scholl

Betreuung:
M.Sc. Seufert

Eine Bachelorarbeit am Lehrstuhl für
Betriebssysteme

ERKLÄRUNG

Hiermit erkläre ich, dass ich diese Abschlussarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen/Hilfsmittel verwendet habe und alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Schriften entnommen wurden, als solche kenntlich gemacht habe. Darüber hinaus erkläre ich, dass diese Abschlussarbeit nicht, auch nicht auszugsweise, bereits für eine andere Prüfung angefertigt wurde.

Inhaltsverzeichnis

1	Motivation	6
1.1	PicoC und RETI	6
1.2	Aufgabenstellung	6
1.3	Eigenheiten der Sprache C	6
1.4	Richtlinien	6
2	Implementierung	7
2.1	Architektur	7
2.2	Lexikalische Analyse	7
2.2.1	Verwendung von Lark	7
2.2.2	Basic Parser	7
2.3	Syntaktische Analyse	7
2.3.1	Verwendung von Lark	7
2.3.2	Umsetzung von Präzidenz	7
2.3.3	Derivation Tree Generierung	8
2.3.4	Early Parser	8
2.3.5	Derivation Tree Vereinfachung	8
2.3.6	Abstrakt Syntax Tree Generierung	8
2.4	Code Generierung	8
2.4.1	Passes	8
2.4.2	Umsetzung von Pointern und Arrays	9
2.4.3	Umsetzung von Structs	9
2.4.4	Umsetzung von Funktionen	9
2.4.5	Umsetzung kleinerer Details	9
2.5	Fehlermeldungen	9
2.5.1	Error Handler	9
3	Ergebnisse und Ausblick	10
3.1	Funktionsumfang	10
3.2	Qualitätssicherung	10
3.3	Kommentierter Kompilervorgang	10
3.4	Erweiterungsideen	10
A	Appendix	11
A.1	Konkrete und Abstrakte Syntax	11
A.2	Bedienungsanleitungen	11
A.2.1	PicoC-Compiler	11
A.2.2	Showmode	11
A.2.3	Entwicklertools	11

Abbildungsverzeichnis

Tabellenverzeichnis

2.1 Präzidenzregeln von PicoC	8
---	---

Definitionen

2.1	Symboltabelle	8
-----	-------------------------	---

1 Motivation

1.1 PicoC und RETI

1.2 Aufgabenstellung

1.3 Eigenheiten der Sprache C

1.4 Richtlinien

2 Implementierung

2.1 Architektur

2.2 Lexikalische Analyse

2.2.1 Verwendung von Lark

2.2.2 Basic Parser

2.3 Syntaktische Analyse

2.3.1 Verwendung von Lark

2.3.2 Umsetzung von Präzidenz

Die **PicoC Sprache** hat dieselben **Präzidenzregeln** implementiert, wie die **Sprache C**¹. Die **Präzidenzregeln** von **PicoC** sind in Tabelle 2.3.2 aufgelistet.

¹[C Operator Precedence - cppreference.com](http://c.operatorprecedence-cppreference.com).

Präzedenz	Operator	Beschreibung	Assoziativität
1	a() a[] a.b	Funktionsaufruf Indexzugriff Attributzugriff	Links, dann rechts →
2	-a !a ~a *a &a	Unäres Minus Logisches NOT und Bitweise NOT Dereferenz und Referenz, auch Adresse-von	Rechts, dann links ←
3	a*b a/b a%b	Multiplikation, Division und Modulo	Links, dann rechts →
4	a+b a-b	Addition und Subtraktion	
5	a<b a<=b a>b a>=b	Kleiner, Kleiner Gleich, Größer, Größer gleich	
6	a==b a!=b	Gleichheit und Ungleichheit	
7	a&b	Bitweise UND	
8	a^b	Bitweise XOR (exclusive or)	
9	a b	Bitweise ODER (inclusive or)	
10	a&& b	Logisches UND	
11	a b	Logisches ODER	
12	a=b	Zuweisung	Rechts, dann links ←
13	a,b	Komma	Links, dann rechts →

Tabelle 2.1: Präzidenzregeln von PicoC

2.3.3 Derivation Tree Generierung

2.3.4 Early Parser

2.3.5 Derivation Tree Vereinfachung

2.3.6 Abstrakt Syntax Tree Generierung

ASTNode

PicoC Nodes

RETI Nodes

2.4 Code Generierung

2.4.1 Passes

PicoC-Shrink Pass

PicoC-Blocks Pass

PicoC-Mon Pass

Definition 2.1: Symboltabelle

RETI-Blocks Pass

RETI-Patch Pass

RETI Pass

2.4.2 Umsetzung von Pointern und Arrays

2.4.3 Umsetzung von Structs

2.4.4 Umsetzung von Funktionen

2.4.5 Umsetzung kleinerer Details

2.5 Fehlermeldungen

2.5.1 Error Handler

3 Ergebnisse und Ausblick

3.1 Funktionsumfang

3.2 Qualitätssicherung

3.3 Kommentierter Kompiliervorgang

3.4 Erweiterungsideen

A Appendix

A.1 Konkrete und Abstrakte Syntax

A.2 Bedienungsanleitungen

A.2.1 PicoC-Compiler

A.2.2 Showmode

A.2.3 Entwicklertools

Literatur

Online

- *C Operator Precedence* - *cppreference.com*. URL: https://en.cppreference.com/w/c/language/operator_precedence (besucht am 27.04.2022).