

Todo list



Fakultät Informatik

---

**Modul (AKRNM),  
(W|S)S-JJJJ[/YYYY]**

## Projekt

Projekttitel oder -kurzbeschreibung

---

**Autor :**

Studierende oder Studierender, Mat.-Nr.: 00000000

**Betreuer:**

Dozentin oder Dozent

Betreuerin oder Betreuer

Salzgitter

Suderburg

Wolfsburg

# Inhaltsverzeichnis

<b>0</b>	<b>Einleitung und Motivation</b>	<b>1</b>
<b>I</b>	<b>Theorie zum Compiler</b>	<b>2</b>
1	Compiler Uebersicht	3
2	Der Top-Down Compiler	4
3	Der Compiler im Zweipass Verfahren	5
4	Der EBNF-basierte Parser	6
5	Die Zwischensprache im Compiler mit Zweipass Verfahren	7
6	Das RISC Binaercode erzeugende Backend	8
<b>II</b>	<b>Umsetzung des Compilers</b>	<b>9</b>
<b>7</b>	<b>Planung Des Compilers</b>	<b>10</b>
7.1	Sektion . . . . .	10
7.1.1	Subsektion . . . . .	10
7.2	Fazit . . . . .	10
<b>8</b>	<b>Umsetzung des Parsers</b>	<b>11</b>
<b>9</b>	<b>Umsetzung der Zwischensprache</b>	<b>12</b>
<b>10</b>	<b>Umsetzung des RISC Backends</b>	<b>13</b>
<b>11</b>	<b>Umsetzung des RISC Interpreters</b>	<b>14</b>
<b>III</b>	<b>Fazit und Ausblick</b>	<b>15</b>
<b>12</b>	<b>Inspektion des Compilers</b>	<b>16</b>
<b>13</b>	<b>Vergleich mit anderen, echten Compilern</b>	<b>17</b>
<b>14</b>	<b>Vorlage</b>	<b>18</b>
14.1	Sektion . . . . .	18
14.1.1	Subsektion . . . . .	18
14.2	Fazit . . . . .	18

<b>15 Schluss und Fazit</b>	<b>19</b>
<b>Literaturverzeichnis</b>	<b>20</b>
<b>A Anhang A</b>	<b>I</b>
<b>B Anhang B</b>	<b>II</b>
<b>C Anhang C</b>	<b>III</b>

# Abbildungsverzeichnis

# Tabellenverzeichnis

# Erklärung

Hiermit versichere ich, die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet zu haben. Ich versichere weiterhin, alle wörtlich oder sinngemäß aus anderen Quellen übernommenen Aussagen als solche gekennzeichnet zu haben. Die eingereichte Arbeit ist weder vollständig noch in wesentlichen Teilen Gegenstand eines anderen Prüfungsverfahrens gewesen.

---

Ort, Datum

---

- Studierende oder Studierender -

# Kapitel 0

## Einleitung und Motivation

Beschreibung der Projektaufgabe

Benennung der Projektziele



**Teil I**

**Theorie zum Compiler**

# Kapitel 1

## Compiler Uebersicht

- Compiler Historie
- Einleitung zu den detaillierteren Beschreibungen:
- Arten von Compilern
  - > Spaeter genauer erklart: Top-Down Compiler
- Compilerphasen
  - Einpass vs Mehrpass vs Zweipass zur Trennung Frontend <-> Backend
  - > Spaeter genauer erklart: Zweipass Compiler fuer separate Backends zukuenftige Optimierungsschritte
- Parser: Arten (EBNF etc)
  - > Spaeter genauer erklart: EBNF
- Zwischensprache: Vereinfacht Optimierungen, separierte Backends und Frontends
  - > Spaeter genauer erklart: ??? (Wie so ein Zwischenstate aussehen soll ist scheinbar schwarze Magie, selber rausfinden)
- Backends: Code Erzeugung etc
  - > Spaeter genauer erklart: RISC Binary, RISC assembly
- Bootstrapping compiler in eigener Sprache
  - > Passiert hier nicht, zu komplex und wild

## Kapitel 2

# Der Top-Down Compiler

Uebersicht und Beschreibung Top-Down Compiler, Verwendung und Software Stacks

## Kapitel 3

# Der Compiler im Zweipass Verfahren

Aufbau und Details zum Zweipass Verfahren

## Kapitel 4

# Der EBNF-basierte Parser

Historie zu EBNF, Verwendungsbeispiele, etc

## Kapitel 5

# Die Zwischensprache im Compiler mit Zweipass Verfahren

Ziele der Zwischensprache, Aufbau, Verwendung, Aufzeigen möglicher Optimierungen

Vielleicht zu wenig Referenz Material fuer Theorie vor der Umsetzung, stattdessen eher die Umsetzung im Fazit Teil der Arbeit dokumentieren?

## Kapitel 6

# Das RISC Binaercode erzeugende Backend

- Aufgabe des Backends
- Beschreibung Zielplattform RISC
  - Detaillierte Uebersicht ueber Assembly, Call Stack und weitere Themen

## Teil II

# Umsetzung des Compilers



# Kapitel 7

## Planung Des Compilers

Die Implementierung des Compilers soll folgende Ziele erfuellen:

- Verstaendlichkeit und Inspizierbarkeit: Der Compiler soll inspizierbar und verstaendlich aufgebaut sein. Zusaetzlich soll die Arbeitsweise des Compilers transparent sein, indem die einzelnen Teile des Compilers gut debugbar oder inspizierbar sind. - Flexibles Backend: Der Compiler soll in der Lage sein, auf verschiedene Zielplattformen wie RISC Assembly und RISC Binary zu kompilieren.

Weitere Aufgaben:

- Die generierte RISC Binary kann irgendwie ausgefuehrt werden

### 7.1 Sektion

Text

#### 7.1.1 Subsektion

Text

### 7.2 Fazit

Text

## Kapitel 8

# Umsetzung des Parsers

Wie sieht die Sprache aus, wie parst der diese Sprache, wie mappe ich das auf die Zwischensprache.

Weitere Details: Speichereffizienz, Geschwindigkeit des Parsers, EBNF trickery (zur Optimierung von Geschwindigkeit etc), Fehlerbehandlung der Input Datei (Syntax Fehler / Logik Fehler / etc)

## Kapitel 9

# Umsetzung der Zwischensprache

Aufbau, wie halte ich Frontends und Backends kompatibel ohne zu sehr auf eine Architektur zu scheitern mit der Datenstruktur

Weitere Details: Speichereffizienz, Geschwindigkeit, Fehlerbehandlung

## Kapitel 10

# Umsetzung des RISC Backends

Wie mappe ich die Zwischensprache auf den Output, wieviel Code kann ich zwischen RISC binary und RISC Assembly sparen

Weitere Details: Speichereffizienz, Geschwindigkeit, Fehlerbehandlung

## Kapitel 11

# Umsetzung des RISC Interpreters

REDACTED: Gibts bereits, ist nicht in Aufgabenstellung, z.B.: <https://www.cs.cornell.edu/courses/cs3410/2018/fall/>  
(=> dafür sorgen das RISC Assembly mit z.B. diesem Interpreter [besser irgendwas open source  
maßes] kompatibel ist)

## Teil III

# Fazit und Ausblick

## Kapitel 12

# Inspektion des Compilers

Performance des Compilers, Bugs?, Debuggability

## Kapitel 13

# Vergleich mit anderen, echten Compilern

Vergleich von Output von Beispiel Code, der gleiches tut, ueber mehrere andere Programmiersprachen und ihre Compiler. Analyse, Ausblick auf Optimierungsmoeglichkeiten.

Vielleicht witzig: Gibt bestimmt einen Compiler Compiler mit RISC Backend, in den ich ein bisl EBNF zum parsen "meiner" Prog Sprache packen kann -> direkter Vergleich zum handgeschriebenen compiler



# Kapitel 14

## Vorlage

Einleitung

### 14.1 Sektion

Text

#### 14.1.1 Subsektion

Text

### 14.2 Fazit

Text

## Kapitel 15

# Schluss und Fazit

Schluss und Fazit

# Literaturverzeichnis

[1] author, *title*, 1st ed. adress: publisher, 0000.

[2] —, *title*. publisher, 0000.

Anhang A

Anhang A

*siehe nächste Seite*

Der Anhang enthält auf den nächsten Seiten [...].

Anhang B

Anhang B

Anhang C

Anhang C