PicoC-Compiler

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

Kolloquiumspräsentation

Präsentator:
Jürgen Mattheis

Gutachter:
Prof. Dr. Scholl

Betreuung:
M.Sc. Seufert

28. September 2022

Universität Freiburg, Lehrstuhl für Betriebssysteme

Code Generierung

Codebeispiel

- im Appendix ab Folie ?? sind Tokens, Ableitungsbaum, Vereinfachter Ableitungsbaum, Abstrakter Syntaxbaum und die modifzierten Abstrakten Synxtaxbäume der verschiedenen Passes an einem Codebeispiel erklärt.
- `> make test TESTNAME=example_presentation VERBOSE=-v`

Code Generierung

Zugriff auf Zusammengesetzte Datentypen

```
1  // in:1
2  // expected:42
3  // datasegment:36
4
5  struct stt {int attr1; int attr2[2];};
6
7  struct stt ar_of_sts[3][2];
8
9  int fun(struct stt (*param)[3][2]){
10   (*(*param+2))[1].attr2[input()] = 42;
11   return 1;
12 }
13
14 void main() {
15   struct stt (*pntr_on_ar_of_sts)[3][2] = &ar_of_sts;
16   int res = fun(pntr_on_ar_of_sts);
17   if (res) {
18     print((*(*pntr_on_ar_of_sts+2))[1].attr2[1]);
19   }
20 }
```

Datentyp der Variable

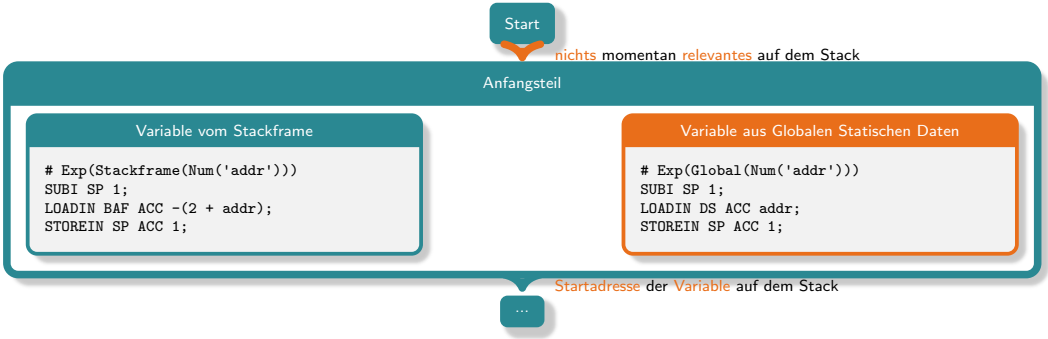
```
// ...
struct stt
↪  (*pntr_on_ar_of_sts)[3][2] =
↪  &ar_of_sts;
// ...
```

- ein „Zeiger auf ein Feld der Mächtigkeit 3 von Feldern der Mächtigkeit 2 von Verbunden des Typs `stt`“.
- *Clockwise/Spiral Rule*

Code Generierung

Codebeispiel

- `(*(*pntr_on_ar_of_sts+2))[1].attr2[1]`



- `(*(*pntr_on_ar_of_sts+2))[1].attr2[1]` `struct stt (*pntr_on_ar_of_sts)[3][2]...`

Mittelteil

Feldindexzugriff auf Feld

```
# z.B. Exp(Num('idx'))
SUBI SP 1;
LOADI ACC idx;
STOREIN SP ACC 1;
# Ref(Subscr(Stack(Num('2')),
↪  Stack(Num('1'))))
LOADIN SP IN1 2;
LOADIN SP IN2 1;
MULTI IN2 (∏_{i=i+1}^{n} dim_j)·size(datatype);
ADD IN1 IN2;
ADDI SP 1;
STOREIN SP IN1 1;
```

Feldindexzugriff auf Zeiger

```
# z.B. Exp(Num('idx'))
SUBI SP 1;
LOADI ACC idx;
STOREIN SP ACC 1;
# Ref(Subscr(Stack(Num('2')),
↪  Stack(Num('1'))))
LOADIN SP IN2 2;
LOADIN IN2 IN1 0;
LOADIN SP IN2 1;
MULTI IN2 (∏_{i=i+1}^{n} dim_j)·size(datatype);
ADD IN1 IN2;
ADDI SP 1;
STOREIN SP IN1 1;
```

Verbundsattribut-zugriff auf Verbund

```
# Ref(Attr(Stack(Num('1')),
↪  Name('attr')))
LOADIN SP IN1 1;
ADDI IN1 Σ_{k=1}^{idx_i-1} size(datatype_{1,k});
STOREIN SP IN1 1;
```

*1

*1: Startadresse eines Zeigerelementes, Feldelementes oder Verbundsattributes auf dem Stack

- `(*(*pntr_on_ar_of_sts+2))[1].attr2[1]` `struct stt (*pntr_on_ar_of_sts)[3][2]...`

...

Startadresse der Variable auf dem Stack

Mittelteil

Feldindexzugriff auf Feld

```
# z.B. Exp(Num('idx'))
SUBI SP 1;
LOADI ACC idx;
STOREIN SP ACC 1;
# Ref(Subscr(Stack(Num('2')),
↪  Stack(Num('1'))))
LOADIN SP IN1 2;
LOADIN SP IN2 1;
MULTI IN2 (∏_{i=i+1}^{n} dim_j)·size(datatype);
ADD IN1 IN2;
ADDI SP 1;
STOREIN SP IN1 1;
```

Feldindexzugriff auf Zeiger

```
# z.B. Exp(Num('idx'))
SUBI SP 1;
LOADI ACC idx;
STOREIN SP ACC 1;
# Ref(Subscr(Stack(Num('2')),
↪  Stack(Num('1'))))
LOADIN SP IN2 2;
LOADIN IN2 IN1 0;
LOADIN SP IN2 1;
MULTI IN2 (∏_{i=i+1}^{n} dim_j)·size(datatype);
ADD IN1 IN2;
ADDI SP 1;
STOREIN SP IN1 1;
```

Verbundsattribut-zugriff auf Verbund

```
# Ref(Attr(Stack(Num('1')),
↪  Name('attr')))
LOADIN SP IN1 1;
ADDI IN1 Σ_{k=1}^{idx_i-1} size(datatype_{1,k});
STOREIN SP IN1 1;
```

*1

*1: Startadresse eines Zeigerelementes, Feldelementes oder Verbundsattributes auf dem Stack

...

- `(*(*pntr_on_ar_of_sts+2))[1].attr2[1]` `struct stt (*pntr_on_ar_of_sts)[3][2]...`

Startadresse der Variable auf dem Stack

Mittelteil

Feldindexzugriff auf Feld

```
# z.B. Exp(Num('idx'))
SUBI SP 1;
LOADI ACC idx;
STOREIN SP ACC 1;
# Ref(Subscr(Stack(Num('2')),
↪  Stack(Num('1'))))
LOADIN SP IN1 2;
LOADIN SP IN2 1;
MULTI IN2 (∏_{i=i+1}^{n} dim_j) · size(datatype);
ADD IN1 IN2;
ADDI SP 1;
STOREIN SP IN1 1;
```

Feldindexzugriff auf Zeiger

```
# z.B. Exp(Num('idx'))
SUBI SP 1;
LOADI ACC idx;
STOREIN SP ACC 1;
# Ref(Subscr(Stack(Num('2')),
↪  Stack(Num('1'))))
LOADIN SP IN2 2;
LOADIN IN2 IN1 0;
LOADIN SP IN2 1;
MULTI IN2 (∏_{i=i+1}^{n} dim_j) · size(datatype);
ADD IN1 IN2;
ADDI SP 1;
STOREIN SP IN1 1;
```

Verbundsattribut-zugriff auf Verbund

```
# Ref(Attr(Stack(Num('1')),
↪  Name('attr')))
LOADIN SP IN1 1;
ADDI IN1 Σ_{k=1}^{idx_i-1} size(datatype_{1,k});
STOREIN SP IN1 1;
```

*1

*1: Startadresse eines Zeigerelementes, Feldelementes oder Verbundsattributes auf dem Stack

- `(*(*pntr_on_ar_of_sts+2))[1].attr2[1]` `struct stt (*pntr_on_ar_of_sts)[3][2]...`

Startadresse der Variable auf dem Stack

Mittelteil

Feldindexzugriff auf Feld

```
# z.B. Exp(Num('idx'))
SUBI SP 1;
LOADI ACC idx;
STOREIN SP ACC 1;
# Ref(Subscr(Stack(Num('2')),
↪  Stack(Num('1'))))
LOADIN SP IN1 2;
LOADIN SP IN2 1;
MULTI IN2 (∏_{i=j+1}^{n} dim_j)·size(datatype);
ADD IN1 IN2;
ADDI SP 1;
STOREIN SP IN1 1;
```

Feldindexzugriff auf Zeiger

```
# z.B. Exp(Num('idx'))
SUBI SP 1;
LOADI ACC idx;
STOREIN SP ACC 1;
# Ref(Subscr(Stack(Num('2')),
↪  Stack(Num('1'))))
LOADIN SP IN2 2;
LOADIN IN2 IN1 0;
LOADIN SP IN2 1;
MULTI IN2 (∏_{i=j+1}^{n} dim_j)·size(datatype);
ADD IN1 IN2;
ADDI SP 1;
STOREIN SP IN1 1;
```

Verbundsattribut-zugriff auf Verbund

```
# Ref(Attr(Stack(Num('1')),
↪  Name('attr')))
LOADIN SP IN1 1;
ADDI IN1 Σ_{k=1}^{idx_j-1} size(datatype_{1,k});
STOREIN SP IN1 1;
```

*1

*1: Startadresse eines Zeigerelementes, Feldelementes oder Verbundsattributes auf dem Stack

- `(*(*pntr_on_ar_of_sts+2))[1].attr2[1],` `struct stt {int attr1; int attr2[2];};`

Startadresse der Variable auf dem Stack

Mittelteil

Feldindexzugriff auf Feld

```
# z.B. Exp(Num('idx'))
SUBI SP 1;
LOADI ACC idx;
STOREIN SP ACC 1;
# Ref(Subscr(Stack(Num('2')),
↪  Stack(Num('1'))))
LOADIN SP IN1 2;
LOADIN SP IN2 1;
MULTI IN2 (∏_{i=j+1}^{n} dim_j) · size(datatype);
ADD IN1 IN2;
ADDI SP 1;
STOREIN SP IN1 1;
```

Feldindexzugriff auf Zeiger

```
# z.B. Exp(Num('idx'))
SUBI SP 1;
LOADI ACC idx;
STOREIN SP ACC 1;
# Ref(Subscr(Stack(Num('2')),
↪  Stack(Num('1'))))
LOADIN SP IN2 2;
LOADIN IN2 IN1 0;
LOADIN SP IN2 1;
MULTI IN2 (∏_{i=j+1}^{n} dim_j) · size(datatype);
ADD IN1 IN2;
ADDI SP 1;
STOREIN SP IN1 1;
```

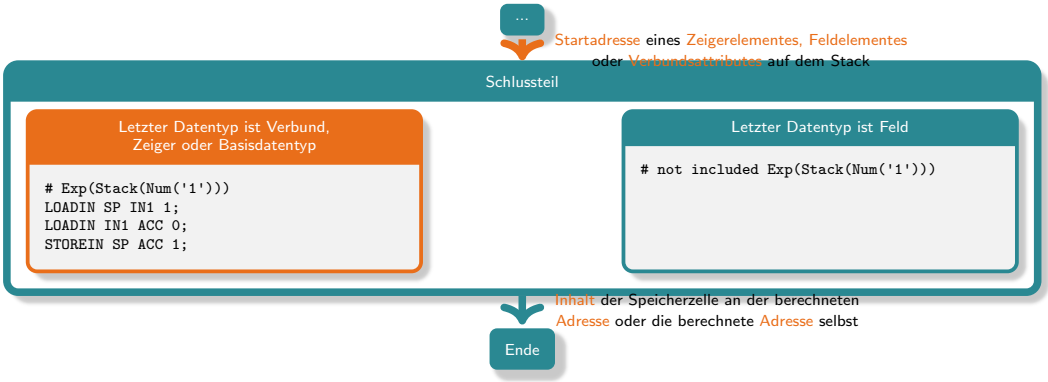
Verbundsattribut-zugriff auf Verbund

```
# Ref(Attr(Stack(Num('1')),
↪  Name('attr')))
LOADIN SP IN1 1;
ADDI IN1 Σ_{k=1}^{idx_i-1} size(datatype_{1,k});
STOREIN SP IN1 1;
```

*1

*1: Startadresse eines Zeigerelementes, Feldelementes oder Verbundsattributes auf dem Stack

- `(*(*pntr_on_ar_of_sts+2))[1].attr2[1],` `struct stt {int attr1; int attr2[2];};`



Fehlermeldungen

Kategorien

- `UnexpectedCharacter`
- `UnexpectedToken`
- `UnexpectedEOF`
- `DivisionByZero`
- `UnknownIdentifier`
- `UnknownAttribute`
- `ReDeclarationOrDefinition`
- `TooLargeLiteral`
- `NoMainFunction`
- `ConstAssign`
- `DatatypeMismatch`
- `PrototypeMismatch`
- `ArgumentMismatch`
- `WrongNumberArguments`
- `WrongReturnType`
- im Appendix ab Folie 3 mit Erklärung.

Literatur

Vorlesungen

Online


Clockwise/Spiral Rule. URL: https://c-faq.com/decl/spiral.anderson.html (besucht am 29. 07. 2022).