

# PROGRAMMIERUNG

## Übung 10: $C_0$ und $AM_0$

Eric Kunze

`eric.kunze@mailbox.tu-dresden.de`

TU Dresden, 21. Juni 2019

# $C_0$ und $AM_0$

- ▶ **Ziel:** Implementierung einer einfachen Programmiersprache  $C_1$
- ▶ **Hier:** nächste Einschränkung auf  $C_0$ 
  - ▷ genau eine main-Funktion
  - ▷ Zugriff auf `stdio` durch `# include`
  - ▷ einzig zugelassene Datenstruktur: `int`, Konstanten
  - ▷ Kontrollstrukturen: Ein-/Ausgabebefehle, Zuweisungen, Sequenzen, Verzweigungen, bedingte Schleifen
- ▶ **Implementierung** durch
  - ▷ Syntax von  $C_0$
  - ▷ Befehle und Semantik einer abstrakten Maschine  $AM_0$
  - ▷ Übersetzer  $C_0 \leftrightarrow AM_0$

# Übersetzung von `if - then - else`

```
sttrans(if (exp) stat1 else stat2, tab, a) :=  
    boolexptrans(exp, tab)  
    JMC a.1;  
    sttrans(stat1, tab, a.2)  
    JMP a.3;  
a.1 : sttrans(stat2, tab, a.4)  
a.3 :
```

für alle  $exp \in W(\langle \text{BoolExpression} \rangle)$ ,  $stat_1, stat_2 \in W(\langle \text{Statement} \rangle)$ ,  $tab \in \text{Tab}$  und  $a \in \mathbb{N}^*$ .

# Aufgabe 1 – Teil (a)

## Baumstrukturierte Adressen:

```
    READ 1;  
    READ 2;  
    LOAD 1;  
    LOAD 2;  
    GT;  
    JMC 1.3.1;  
    LOAD 1;  
    STORE 3;  
    JMP 1.3.3;  
1.3.1:  LOAD 2;  
        STORE 3;  
1.3.3:  WRITE 3;
```

## Linearisierte Adressen:

```
1:  READ 1;  
2:  READ 2;  
3:  LOAD 1;  
4:  LOAD 2;  
5:  GT;  
6:  JMC 10;  
7:  LOAD 1;  
8:  STORE 3;  
9:  JMP 12;  
10: LOAD 2;  
11: STORE 3;  
12: WRITE 3;
```

# Aufgabe 1 – Teil (b)

## Ablauf der abstrakten Maschine:

	BZ	,	DK	,	HS	,	Inp	,	Out
(	1	,	$\varepsilon$	,	[ ]	,	5:7	,	$\varepsilon$
(	2	,	$\varepsilon$	,	[1/5]	,	7	,	$\varepsilon$
(	3	,	$\varepsilon$	,	[1/5, 2/7]	,	$\varepsilon$	,	$\varepsilon$
(	4	,	5	,	[1/5, 2/7]	,	$\varepsilon$	,	$\varepsilon$
(	5	,	7:5	,	[1/5, 2/7]	,	$\varepsilon$	,	$\varepsilon$
(	6	,	0	,	[1/5, 2/7]	,	$\varepsilon$	,	$\varepsilon$
(	10	,	$\varepsilon$	,	[1/5, 2/7]	,	$\varepsilon$	,	$\varepsilon$
(	11	,	7	,	[1/5, 2/7]	,	$\varepsilon$	,	$\varepsilon$
(	12	,	$\varepsilon$	,	[1/5, 2/7, 3/7]	,	$\varepsilon$	,	$\varepsilon$
(	13	,	$\varepsilon$	,	[1/5, 2/7, 3/7]	,	$\varepsilon$	,	7

$$\mathcal{P}[\llbracket Max_0 \rrbracket](5 : 7) = proj_5^{(5)}\left(I[\llbracket Max_0 \rrbracket](1, \varepsilon, [], 5 : 7, \varepsilon)\right) = 7$$

## Aufgabe 2 – Teil (a)

1: READ 2;	10: LIT 1;
2: READ 3;	11: ADD;
3: LIT 0;	12: STORE 1;
4: STORE 1;	13: LOAD 2;
5: LOAD 1;	14: LOAD 2;
6: LOAD 3;	15: MUL;
7: LT;	16: STORE 2;
8: JMC 18;	17: JMP 5;
9: LOAD 1;	18: WRITE 2;

## Aufgabe 2 – Teil (b)

	BZ	,	DK	,	HS	,	Inp	,	Out
(	1	,	$\varepsilon$	,	[ ]	,	0:1	,	$\varepsilon$
(	2	,	$\varepsilon$	,	[1/0]	,	1	,	$\varepsilon$
(	3	,	$\varepsilon$	,	[1/0, 2/1]	,	$\varepsilon$	,	$\varepsilon$
(	4	,	0	,	[1/0, 2/1]	,	$\varepsilon$	,	$\varepsilon$
(	5	,	1:0	,	[1/0, 2/1]	,	$\varepsilon$	,	$\varepsilon$
(	6	,	0:1:0	,	[1/0, 2/1]	,	$\varepsilon$	,	$\varepsilon$
(	7	,	1:0	,	[1/0, 2/1]	,	$\varepsilon$	,	$\varepsilon$
(	8	,	0	,	[1/0, 2/1]	,	$\varepsilon$	,	$\varepsilon$
(	5	,	0	,	[1/0, 2/1]	,	$\varepsilon$	,	$\varepsilon$
(	6	,	0:0	,	[1/0, 2/1]	,	$\varepsilon$	,	$\varepsilon$
(	7	,	0	,	[1/0, 2/1]	,	$\varepsilon$	,	$\varepsilon$
(	9	,	$\varepsilon$	,	[1/0, 2/1]	,	$\varepsilon$	,	$\varepsilon$
(	10	,	$\varepsilon$	,	[1/0, 2/1]	,	$\varepsilon$	,	1