

# Aufgabe 2a

Freitag, 26. Juni 2020

## Aufgabe 2 (AGS 15.16 a, AGS 15.18 b)

(a) Gegeben sei folgendes Fragment eines C<sub>1</sub>-Programms:

```
#include <stdio.h>
int x, y;
void f(...) {...}

void g(int a, int *b) {
    int c;
    c = 3;
    if (c == *b) while (a > 0) f(&a, b);
}

void main() {...}
```

Übersetzen Sie die Sequenz der Statements im Rumpf von g in entsprechenden AM<sub>1</sub>-Code mit baumstrukturierten Adressen (mittels *stseqtrans*). Sie brauchen keine Zwischenschritte anzugeben. Geben Sie zunächst die benötigte Symboltabelle *tab<sub>g</sub>* an.

$$\begin{array}{|c|c|c|c|c|} \hline a & b & ra & par & c \\ \hline -3 & -2 & -1 & 0 & 1 \\ \hline \end{array}$$

$$adr(r, b, o) = \begin{cases} rto & \text{lokal} \\ o & \text{global} \end{cases}$$

$tab_g = [ f / (proc, 1), g / (proc, 2), x / (var, global, 1), y / (var, global, 2), a / (var, lokal, -3), b / (var-ref, -2), c / (var, lokal, 1) ]$

LIT 3 ; STORE(lokal, 1)

LOAD(lokal, 1) ; LOADI(-2) ; EQ ; JMC 2.2.1

2.2.2.1 : LOAD(lokal, -3) ; LIT 0 ; GT ; JMC 2.2.2.2

LOAD(A)(lokal, -3) ; PUSH ; LOAD(lokal, -2) ; PUSH ; CALL 1 ;  
JMP 2.2.2.1

2.2.2.2 : 2.2.1 :

# Aufgabe 2b

Freitag, 26. Juni 2020

(b) Gegeben sei folgender  $AM_1$ -Code:

1: INIT 1;	8: LOADI(-2);	15: LOADA(global, <u>1</u> );
2: CALL 13;	9: LIT 2;	16: PUSH;
3: INIT 0;	10: DIV;	17: CALL 3;
4: LOADI(-2);	11: STOREI(-2);	18: WRITE(global, 1);
5: LIT 2;	12: RET 1;	19: JMP 0;
6: GT;	13: INIT 0;	
7: JMC 12;	14: READ(global, <u>1</u> );	

Erstellen Sie ein Ablaufprotokoll der  $AM_1$ , indem Sie sie schrittweise ablaufen lassen, bis die Maschine terminiert. Die Anfangskonfiguration sei  $(14, \varepsilon, 0 : 0 : 1, 3, 4, \varepsilon)$ . Sie müssen nur Zellen ausfüllen, deren Wert sich im Vergleich zur letzten Zeile geändert hat.

BZ	DK	LZK	REF	Inp	Out
14	$\varepsilon$	0 : 0 : 1	3	4	$\varepsilon$
15	$\varepsilon$	4 : 0 : 1	3	$\varepsilon$	$\varepsilon$
16	(1)	4 : 0 : 1	3	$\varepsilon$	$\varepsilon$
17	$\varepsilon$	4 : 0 : 1 : 1	3	$\varepsilon$	$\varepsilon$
3	$\varepsilon$	4 : 0 : 1 : 1 : 18 : 3	6	$\varepsilon$	$\varepsilon$
4	$\varepsilon$	(4) 0 : 1 : 1 : 18 : 3	6	$\varepsilon$	$\varepsilon$
5	4	4 : 0 : 1 : 1 : 18 : 3	6	$\varepsilon$	$\varepsilon$
6	2 : 4	4 : 0 : 1 : 1 : 18 : 3	6	$\varepsilon$	$\varepsilon$
7	1	4 : 0 : 1 : 1 : 18 : 3	6	$\varepsilon$	$\varepsilon$
8	$\varepsilon$	(4) 0 : 1 : 1 : 18 : 3	6	$\varepsilon$	$\varepsilon$
9	4	4 : 0 : 1 : 1 : 18 : 3	6	$\varepsilon$	$\varepsilon$
10	2 : 4	4 : 0 : 1 : 1 : 18 : 3	6	$\varepsilon$	$\varepsilon$
11	(2) <sup>4:2=2</sup>	4 : 0 : 1 : 1 : 18 : 3	6	$\varepsilon$	$\varepsilon$
12	$\varepsilon$	2 : 0 : 1 : 1 : (18) (3)	6	$\varepsilon$	$\varepsilon$
18	$\varepsilon$	(2) : 0 : 1	3	$\varepsilon$	$\varepsilon$
19	$\varepsilon$	2 : 0 : 1	3	$\varepsilon$	2
0	$\varepsilon$	2 : 0 : 1	3	$\varepsilon$	2

↑  
 Rücksprung-  
 adresse  
 previous  
 activation  
 record