
Programmierung

Aufgabe 1 (AGS 15.17)

(a) Gegeben sei folgendes Fragment eines C_1 -Programms mit den Funktionen f und g :

```
while (*p > i) { f(p); i = i + 1; }
p = &i;
```

Übersetzen Sie die Sequenz dieser Statements in entsprechenden AM_1 -Code mit baumstrukturierten Adressen (mittels *stseqtrans*). Sie müssen keine Zwischenschritte angeben. Nehmen Sie an, die `while`-Anweisung sei das zweite Statement in g , und es sei

$$tab_{g+lDecl} = \{f/(proc, 1), g/(proc, 2), i/(var, lokal, 1), p/(var-ref, -2)\} .$$

(b) Gegeben sei folgender AM_1 -Code:

```
1  INIT 1;           10  MUL;           19  READ(global,1);
2  CALL 18;         11  STOREI(-3);    20  LOADA(global,1);
3  INIT 0;          12  LOAD(lokal,-2); 21  PUSH;
4  LOAD(lokal,-2);  13  LIT 1;         22  LOAD(global,1);
5  LIT 0;           14  SUB;           23  PUSH;
6  GT;              15  STORE(lokal,-2); 24  CALL 3;
7  JMC 17;          16  JMP 4;         25  WRITE(global,1);
8  LIT 2;           17  RET 2;         26  JMP 0;
9  LOADI(-3);       18  INIT 0;
```

Führen Sie 11 Schritte der AM_1 auf der Konfiguration $\sigma = (22, \varepsilon, 1 : 3 : 0 : 1, 3, \varepsilon, \varepsilon)$ aus.

Aufgabe 2 (AGS 15.16 a)

Gegeben sei folgendes Fragment eines C_1 -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_1 -Code mit baumstrukturierten Adressen (mittels *stseqtrans*). Sie brauchen keine Zwischenschritte anzugeben. Geben Sie zunächst die benötigte Symboltabelle *lokal-tab_g* an.

Aufgabe 3 (AGS 15.18 b)

Gegeben sei folgender AM_1 -Code:

1 INIT 1;	8 LOADI(-2);	15 LOADA(global, 1);
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, 1);	

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.

Zusatzaufgabe 1 (AGS 13.11 b,c)

Ein Binärbaum sei durch ein ternäres Funktionssymbol $\text{tree}(V, \text{Left}, \text{Right})$ gegeben, wobei V eine Knotenbeschriftung, Left der linke Kindbaum und Right der rechte Kindbaum ist. Das Funktionssymbol nil wird für den leeren Baum verwendet.

(a) Geben Sie den Baum
$$\begin{array}{c} 1 \\ / \quad \backslash \\ 2 \quad 3 \\ / \quad \backslash \\ 4 \quad 5 \end{array}$$
 als Term in Prolog an. Verwenden Sie dazu tree und nil .

(b) Geben Sie eine binäre Relation yield an, die genau die Paare von Bäumen und Listen (ξ, v) enthält, sodass v nur die Blattbeschriftungen in ξ von links nach rechts enthält.