

1. Übung

Fakultät Verkehrswissenschaften Fachrichtung Verkehrsingenieurwesen

Zeitraum: 11. bis 21.04.2011

Aufgabe 1: Ausführen von AM_0 -Programmen

(a) Folgende linearisierte Übersetzung $prog_0$ sei gegeben:

| | | |
|-------------|--------------|--------------|
| 1: LIT 1; | 7: JMC 17; | 13: LIT 1; |
| 2: STORE 2; | 8: LOAD 2; | 14: SUB; |
| 3: READ 1; | 9: LOAD 1; | 15: STORE 1; |
| 4: LOAD 1; | 10: MUL; | 16: JMP 4; |
| 5: LIT 1; | 11: STORE 2; | 17: WRITE 2; |
| 6: GT; | 12: LOAD 1; | |

Berechnen Sie $\mathcal{P}[\![prog_0]\!](2)$ durch Ablaufenlassen der AM_0 .

Dokumentieren Sie den Zustand der AM_0 nach Ausführung jedes Befehls.

(b) Gegeben sei das folgende AM_0 -Programm $prog_0$:

| | | | |
|------------|------------|-------------|------------|
| 1: READ 1; | 4: GT; | 7: LIT 2; | 10: JMP 2; |
| 2: LOAD 1; | 5: JMC 11; | 8: DIV; | |
| 3: LIT 0; | 6: LOAD 1; | 9: STORE 1; | |

Führen Sie $prog_0$ auf der AM_0 mit der Anfangskonfiguration $(1, \varepsilon, [], 1, \varepsilon)$ schrittweise aus, bis eine Endkonfiguration erreicht ist.

Aufgabe 2: Übersetzung von C_0 -Ausdrücken

Gegeben sei die folgende, durch die Übersetzung des Deklarationsteiles eines C_0 -Programms entstandene, Symboltabelle:

| id | a | x | y | z |
|---------|------------|----------|----------|----------|
| tab(id) | (const, 7) | (var, 1) | (var, 2) | (var, 3) |

Übersetzen Sie die folgenden Ausdrücke in AM_0 -Befehlsfolgen:

- $a * x + z$
- $y * 2 - z/a$
- $(a * x + z) * 3$
- $x * (z - y \% 2)$

Aufgabe 3: Übersetzung und Ausführung eines Programms

Gegeben sei das folgende C_0 -Programm *filter* :

```
#include <stdio.h>

int main()
{ int x;

  scanf("%i",&x);
  while(x!=0)
    { if (x%3 == 0) printf("%d",x);
      scanf("%i",&x);
    }
  return 0;
}
```

- (a) Erzeugen Sie aus dem C_0 -Programm *filter* ein AM_0 -Programm *filter*₀ durch Übersetzen mittels Transformationskalkül *trans*.
- Hinweis:** Erzeugen Sie zunächst ein Programm *bfilter*₀ mit baumstrukturierten Adressen und daraus durch Linierisierung der Adressen das Programm *filter*₀ .
- (b) Berechnen Sie $\mathcal{P}[[filter_0]](6.4.3.0)$ durch Ablaufenlassen auf der AM_0 .
- Hinweis:** Der Punkt zwischen 6, 4, 3 und 0 ist **kein** Dezimalpunkt, sondern das Trennzeichen zwischen den eingegebenen Werten!