

Aufgabenblatt zur 8. Übung

Zeitraum: 08.06. bis 12.06.2009

1. Aufgabe: (AGS 12.6* a)

(a) Gegeben sei folgendes C_0 -Programm:

```
/*ggT*/
#include <stdio.h>

int main()
{ int a,b,h;
  scanf("%i",&a);
  scanf("%i",&b);
  if (b>0) {
    while (b>0)
      {h=a % b; a=b; b=h;}
    printf("%d",a);
  }
  return 0;
}
```

Übersetzen Sie obiges C_0 -Programm in ein AM_0 -Programm ggT_0 mit linearisierten Adressen. Zwischenschritte brauchen Sie nicht anzugeben.

2. Aufgabe: (AGS 12.7* a)

Gegeben sei folgendes C_0 -Programm:

```
/* prog */
#include <stdio.h>
int main() {
  int x, y, z;
  z = 1;
  scanf("%i", &x);
  scanf("%i", &y);
  while (x > y) {
    z = y * z;
    x = x - 1;
  }
  printf("%d", z);
  return 0;
}
```

Übersetzen Sie obiges C_0 -Programm in ein AM_0 -Programm $prog_0$ mit linearisierten Adressen. Zwischenschritte brauchen Sie keine anzugeben.

3. Aufgabe: (AGS 13.5 b,c)

(b) Das Programm $prog_1$ sei durch folgenden AM_1 -Code gegeben:

```
1:  INIT 2;           8:  STORE(global, 1);   15:  LOADA(global,2);
2:  CALL 10;         9:  RET 1;             16:  PUSH;
3:  JMP 0;           10: INIT 1;           17:  CALL 4;
4:  INIT 1;          11: READ(lokal, 1);    18:  WRITE(lokal, 1);
5:  LOADI(-2);       12: LOAD(lokal, 1);    19:  RET 0;
6:  STORE(lokal, 1); 13: STORE(global, 1);
7:  LIT 20;          14: READ(global, 2);
```

Berechnen Sie die Programmsemantik $\mathcal{P}[[prog_1]](5 : 8)$

(c) Stellen Sie den Aufbau des Laufzeitkellers für die Befehlszeile 7 graphisch dar. Markieren Sie dabei die Aktivierungsblöcke und die maßgeblichen Abhängigkeiten innerhalb des Laufzeitkellers.

Zusatzaufgabe: (AGS 12.1*)

Gegeben sei folgendes C_0 -Programm Max :

```
/* Max */
#include <stdio.h>
int main() {
    int a, b, max;
    scanf("%i", &a);
    scanf("%i", &b);
    if (a > b) max = a;
    else max = b;
    printf("%d", max);
    return 0;
}
```

(a) Berechnen Sie schrittweise das baumstrukturierte Programm $bMax_0 = \underline{trans}(Max)$ mit Hilfe der in der Vorlesung/Skript angegebenen Übersetzungsfunktionen.

(b) Wandeln Sie $bMax_0$ in ein Programm Max_0 mit linearisierten Adressen um und berechnen Sie $\mathcal{P}[[Max_0]](5 : 7)$. Dokumentieren Sie den Zustand der AM_0 nach Ausführung jedes Befehls.