

Übung 08 zum Compilerbau WS14/15 bis: 09.12.2014**Aufgabe 37:**

Berechnen Sie die Semantik $M(P)$ Ihres Mini-Java-Programms aus Praktikum Versuch 1 Aufgabe 1.

Aufgabe 38:

- a) Erweitern Sie die Definition der Semantik (Statement-Semantik S) von Mini-Java um die beiden in Aufgabe 32 beschriebenen Java-Konstrukte.
- b) Berechnen Sie die Semantik $M(P)$ des folgenden Programms P:

```
int out, i=1;
{
  do {
    switch i {
      case 1: out=1;
      case 2: out=2;
      default: out=0;
    }
    i=i+1;
  } while i < 2
  print(out);
}
```

Aufgabe 39:

- a) Geben Sie die Übersetzung des Mini-Java-Programms aus der Aufgabe 37 in Z-Mini-Java-Code an. Nutzen Sie dazu die aus der Vorlesung bekannten Übersetzungsfunktionen.
- b) Führen Sie die Rechnung des Z-Mini-Java-Codes aus a) schrittweise durch und vergleichen Sie das Ergebnis mit dem Ergebnis aus Aufgabe 37.

Aufgabe 40:

- a) Erweitern Sie die Definition der Übersetzung von Mini-Java in Z-Mini-Java-Code um die beiden in Aufgabe 32 beschriebenen Java-Konstrukte.
- b) Führen Sie die Übersetzung des Mini-Java-Programms aus Aufgabe 32 in Z-Mini-Java-Code durch.
- c) Führen Sie die Rechnung des Z-Mini-Java-Codes aus b) schrittweise durch und vergleichen Sie das Ergebnis mit dem Ergebnis aus Aufgabe 38 b).

Aufgabe 41:

Die große Ackermannfunktion ist wie folgt definiert:

```
ack: nat x nat -> nat
ack (a,0) = 1, für alle nat. Zahlen a >= 0
ack (0,1) = 2
ack (0, b+2) = b + 4, für alle nat. Zahlen b >= 0
ack (a+1,b+1) = ack (a,ack(a+1,b)), a,b >= 0
```

Berechnen Sie die Funktionswerte:

```
ack(0,0), ack(0,1), ..., ack(0,14)
ack(1,0), ack(1,1), ..., ack(1,8)
ack(2,0), ack(2,1), ..., ack(2,4)
ack(3,0), ack(3,1), ..., ack(3,3)
```

und tragen Sie sie in folgendes Schema ein:

Übung 08 zum Compilerbau WS14/15 bis: 09.12.2014

A / B	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0															
1															
2															
3															

Aufgabe 42:

Geben Sie ein M32-Assembler-Programm zur Implementierung der großen Ackermannfunktion an, wobei Sie die Eingabewerte als Konstanten im Programm definieren.

Aufgabe 43:

Geben Sie ein Mini-Java-Programm zur Implementierung der großen Ackermannfunktion an, wobei Sie die Eingabewerte als Konstanten im Programm definieren.