

Übungen zur Semantik von Programmiersprachen

Aufgabe 1 (H) (*Totale Korrektheit*) (Abgabe: Montag, 4. 2. 2002)

Gegeben sei folgendes Programm:

```
{ 0 ≤ N0 }  
  N := 0; P := 1; C := 0;  
  while ¬(N = N0) do  
    if C = 0 then (N := N + 1; C := P)  
    else (C := C - 1; P := P + 1);  
  P := P + C  
{ P = ? }
```

- (a) Was ist der Wert von P nach Ausführung des Programms?
- (b) Wie lautet die Invariante der Schleife?
- (c) Zeigen Sie die totale Korrektheit des Programms.

Benutzen Sie dazu folgende Regel für die While-Schleife:

$$\frac{\{I \wedge b \wedge t = z\} s \{I \wedge t < z\}}{\{I\} \mathbf{while} b \mathbf{do} s \{I \wedge \neg b\}}$$

Der Term t darf beliebige aus der Mathematik bekannte Funktionen enthalten.

Aufgabe 2 (Ü) (*Totale Korrektheit*)

In dieser Aufgabe soll noch einmal die totale Korrektheit des Programms aus Aufgabe 1 gezeigt werden. Diesmal jedoch mit der modifizierten While-Regel:

$$\frac{P(n+1) \implies b \quad \{P(n+1)\} s \{P n\} \quad P 0 \implies \neg b}{\{\exists n. P n\} \mathbf{while} b \mathbf{do} s \{P 0\}}$$

Zusätzlich soll diesmal die syntaktisch eingeschränkte Version der Assertions benutzt werden. Das bedeutet insbesondere, dass nur die Funktionen $+$ und $*$ vorkommen dürfen.

Aufgabe 3 (Ü) (*Zeitkomplexität*)

```
n = 0; m = 0; k = 0;  
while k < a do  
  (n = n + 1; k = k + m + 1; m = m + 2)
```

Beweisen Sie die totale Korrektheit des Programms und zeigen Sie mit Hilfe des Erweiterten Hoare-Kalküls, daß die Ausführungszeit des Programms höchstens $\mathcal{O}(\sqrt{a})$ ist.

Aufgabe 4 (Ü) (*Arrays*)

Zeigen Sie die Gültigkeit folgenden Hoare-Tripels:

$$\{A[I] = I\} A[A[I]] := I \{A[I] = I\}$$

Aufgabe 5 (P) (*Zeitkomplexität*)

(Abgabe: Montag, 4. 2. 2002 an kleing@in.tum.de)

Erweitern Sie die Big-Step-Semantik der Sprache WHILE in Prolog um die Berechnung der exakten Laufzeit des Programms.

Sie können dazu die Lösung der Programmieraufgabe von Blatt 2 verwenden.