

Übungen zur Semantik von Programmiersprachen

Aufgabe 1 (H) (*Monotonie von pre und vc*) **Abgabe: Montag, 28. 1. 2002**

Zeigen Sie, daß die in der Vorlesung definierten Funktionen *pre* und *vc* monoton bezüglich der Implikation sind, d.h. dass gilt:

- (a) $\forall s \in \mathbf{Stmt}, B, C \in \mathbf{Assn} : (B \Rightarrow C) \Rightarrow (pre\ s\ B \Rightarrow pre\ s\ C)$
- (b) $\forall s \in \mathbf{Stmt}, B, C \in \mathbf{Assn} : (B \Rightarrow C) \Rightarrow (vc\ s\ B \Rightarrow vc\ s\ C)$

Hinweis: Strukturelle Induktion über *s*. Sie dürfen verwenden, dass die Substitution monoton bezüglich der Implikation ist, d.h. daß für alle $a \in \mathbf{Aexp}$ und Programmvariablen *X* gilt: $\forall B, C \in \mathbf{Assn} : (B \Rightarrow C) \Rightarrow (B[a/X] \Rightarrow C[a/X])$.

Aufgabe 2 (Ü) (*Prozeduren im Hoare-Kalkül*)

Seien *X, Y, Z* natürlichzahlige Programmvariablen. Gegeben ist folgende WHILE-Prozedur:

```
proc mult = if Y = 0
              then Z := 0
              else (Y := Y - 1 ; call mult ; Y := Y + 1 ; Z := Z + X)
```

Beweisen Sie in der Hoare-Logik: $\{\mathbf{true}\} \text{ call mult } \{Z = X \cdot Y\}$.

Aufgabe 3 (Ü) (*Verifikationsbedingungen*)

Die Funktion *vc* aus der Vorlesung stützt sich bei der Berechnung von Verifikationsbedingungen ausschließlich auf ein Statement *s* und eine Nachbedingung *B*, so daß

$$vc\ s\ B \Rightarrow \models \{pre\ s\ B\} s \{B\}$$

- (a) Geben Sie eine analoge Funktion *vc'* an, die Verifikationsbedingungen zu vollständigen Hoare-Tripeln $\{A\} s \{B\}$ berechnet.
- (b) Zeigen Sie die Korrektheit von *vc'* in Bezug auf die Funktionen *pre* und *vc* aus der Vorlesung.

Aufgabe 4 (P) (*Generator für Verifikationsbedingungen in Gofer*)

Abgabe: Montag, 28. 1. 2002 an kleing@in.tum.de

Implementieren Sie den Generator für Verifikationsbedingungen aus der Vorlesung in der Programmiersprache Gofer. Einen Rahmen für das Programm finden Sie in der Datei

`/usr/proj/semantik/prog/gofer/vc-gen/rahmen.gs`