

“Analyse und Verifikation (185.276, VU 2.0, ECTS 3.0)”

SS 2009

Übungsblatt 1

17.03.2009

Aufgabe 1 : (8+2 Punkte)

Unter den *freien Variablen* eines Booleschen Ausdrucks b verstehen wir die Menge der in ihm vorkommenden Variablen. Diese Menge lässt sich induktiv wie folgt definieren:

$$\begin{aligned}FV(true) &= \emptyset \\FV(false) &= \emptyset \\FV(a_1 = a_2) &= FV(a_1) \cup FV(a_2) \\FV(a_1 \leq a_2) &= FV(a_1) \cup FV(a_2) \\FV(\neg b_1) &= FV(b_1) \\FV(b_1 \wedge b_2) &= FV(b_1) \cup FV(b_2) \\FV(b_1 \vee b_2) &= FV(b_1) \cup FV(b_2)\end{aligned}$$

1. Beweisen Sie induktiv: Sind σ und σ' zwei Zustände mit $\sigma(x) = \sigma'(x)$ für alle $x \in FV(b)$, dann gilt:

$$\llbracket b \rrbracket_B(\sigma) = \llbracket b \rrbracket_B(\sigma')$$

2. Was bedeutet die vorstehende Aussage anschaulich?

Aufgabe 2 : (8+2 Punkte)

Gegeben sei das folgende WHILE-Programm π :

```
z:=0; while y<=x do z:=z+1; x:=x-y od
```

1. Geben Sie die Ableitungsfolge an, die sich ergibt, wenn das Programm π auf einen Zustand σ mit $\sigma(x) = 19$ und $\sigma(y) = 6$ angesetzt wird.
2. Geben Sie einen Zustand τ an, für den π angesetzt auf τ divergiert.

Aufgabe 3 : (5 Punkte)

In dieser Aufgabe erweitern wir die Sprache WHILE um das Konstrukt

```
repeat  $\pi$  until  $b$  taeper
```

Geben Sie eine SOS-Regel [rep_sos] an, die diesem Konstrukt die “gewohnte” Semantik gibt, ohne bei dieser Angabe die Existenz des while-Konstrukts in WHILE auszunutzen.

Abgabe: Dienstag, den 31.03.2009, vor der Vorlesung (HS 4, Freihaus).