

Übungen zu „Semantik von Programmiersprachen“, Sommer 2010

Nr. 9, Abgabe der Aufgaben: 22. Juni 2010 vor der Vorlesung

Aufgaben

9.1 Leiten Sie mit Hilfe der Hoare-Regeln her:

6 Punkte

```
{X = n ∧ Y = m ∧ Z = 1}
while ¬(Y = 0) do (
  while even(Y) do (X := X * X; Y := Y/2);
  Z := Z * X;
  Y := Y - 1
)
{Z = n^m}
```

Dabei verwenden wir ganzzahlige Division als Erweiterung der arithmetischen Ausdrücke sowie ein Prädikat *even*, das angibt, ob eine Zahl gerade ist.

9.2 **Schwächste Vorbedingung**

6 Punkte

Die Anweisung $c \equiv \mathbf{while} \neg(X = 1) \mathbf{do} (Y := Y * X; X := X - 1)$ besitzt bekanntlich die folgende Semantik:

$$\mathcal{C}[[c]]\sigma = \begin{cases} \sigma[Y \mapsto \sigma(Y) \cdot \sigma(X)!, X \mapsto 1] & : \sigma(X) > 0 \\ \perp & : \sigma(X) \leq 0 \end{cases}$$

Wir betrachten die Nachbedingung $B \equiv Y = i! \wedge i \geq 0$.

- (a) Begründen Sie: $(X = i \wedge Y = 1)^I \neq \mathbf{wp}^I[[c, B]]$ / 1
- (b) Finden Sie eine Zusicherung C , welche äquivalent zur schwächsten Vorbedingung der Zusicherung ist. Beweisen Sie dazu: / 5

$$\forall I \in \mathcal{I} : C^I = \mathbf{wp}^I[[c, B]].$$