

4. Übungsblatt

Ausgabe: 04.05.17

Abgabe: 11.05.17

4.1 Maximale Korrektheit?

5 Punkte

Folgendes Stück Code soll in z das Maximum der beiden Variablen x und y berechnen:

```
// {???\nif (x ≥ y) z = x; else z = y;\n// {???
```

Geben Sie eine geeignete Spezifikation an (ersetzen Sie ??? durch geeignete Vor- und Nachbedingungen), und zeigen Sie mit dem Hoare-Kalkül, dass das Programm die Spezifikation erfüllt. Die Spezifikation sollte vollständig sein, d.h. andere Programme, die nicht das Maximum berechnen, ausschließen.

Geben Sie den Beweis durch vollständige Annotation des Programmes in der in Vorlesung 2 eingeführten Notation an.

4.2 Quadratisch, praktisch, schnell

10 Punkte

Nebenstehendes Stück C0 soll das Quadrat einer ganzen Zahl x berechnen. Wir prüfen das erst einmal exemplarisch: geben Sie dazu die Auswertung des Programmes mittels operationaler Semantik für $\sigma(x) = 2$ an. (Sie können hier die Auswertung der Ausdrücke abkürzen; geben Sie nur die Herleitung der Zustandsübergänge \rightarrow_{stmi} an.) Wenn die Auswertung den gewünschten Wert ergibt, beweisen Sie dann mit dem Hoare-Kalkül in der Notation aus Vorlesung 2 die partielle Korrektheit des Programmes.

```
// {x = X}\np = 0;\nwhile (x ≠ 0) {\n    x = x - 1;\n    p = p + 2 * x + 1;\n}\n// {p = X * X}
```

4.3 Partielle Korrektheit

5 Punkte

Folgendes Stück C0 terminiert nicht für Zustände mit $\sigma(x) > 50$.

```
while (x ≠ 50) {\n    x = x + 1;\n}
```

Dennoch läßt sich etwas über das Programm beweisen. Beweisen Sie die folgenden spezifizierten partiellen Korrektheitsaussagen mit dem Hoare-Kalkül in der Notation aus Vorlesung 2.

```
// {x ≤ 50}\nwhile (x ≠ 50) {\n    x = x + 1;\n}\n// {x = 50}
```

```
// {x > 50}\nwhile (x ≠ 50) {\n    x = x + 1;\n}\n// {x > 50}
```