WS 2020 / 2021 13.11.2020

Übungen zur Vorlesung
Modern Concurrency Theory
Blatt 3

Prof. Dr. Roland Meyer
Sebastian Wolff Abgabe bis 23.11.2020 um 10:00 Uhr

Aufgabe 3.1 (Hoare Kalkül)

Beweisen Sie $\vdash \{b = x \wedge y = 0 \wedge x \geq 0\}\ w\ \{x = y\}$ im Hoare Kalkül, wobei

$$w := \texttt{while}\ (b \neq 0)\ \texttt{do}\ y := y + 1;\ b := b - 1\ \texttt{end}.$$

Aufgabe 3.2 (Schwächste Liberale Precondition)

Sei $\mathcal{S}[\![A]\!] = wlp(c, B)$. Zeigen Sie, dass folgende Aussagen gelten:

1. $\{A\}\ c\ \{B\}$ ist gültig.
2. Falls $\{A'\}\ c\ \{B\}$ gültig ist, dann gilt $A' \Rightarrow A$.

Aufgabe 3.3 (Unentscheidbarkeit)

Zeigen Sie informell, dass die Verifikation von `W--` Programmen unentscheidbar ist. Nehmen Sie dazu an, dass das Prüfen der Gültigkeit von Hoare-Tripeln berechenbar ist.

Hinweis: Zeigen Sie zuerst die Turing-Vollständigkeit von `W--`*. Dazu können Sie Zählermaschinen statt Turingmaschinen betrachten. Leiten Sie danach einen Widerspruch her, indem Sie ein Entscheidungsverfahren für das Haltproblem angeben. Dazu müssen Sie Terminierung mittels Hoare-Tripeln ausdrücken.*

Aufgabe 3.4 (Verification Conditions)

Betrachten Sie folgendes (annotiertes) Programm p.

$\{A\} = \{a \geq 0 \wedge c = 0 \wedge x = a\}$
while $(a > 0)$ **do**
  $\{I_1\} = \{c = \sum_{i=a+1}^{x} i \wedge a \geq 0 \wedge x \geq 0\}$
  $b := 0$
  while $(a \neq b)$ **do**
    $\{I_2\} = \{c = b + \sum_{i=a+1}^{x} i \wedge a > 0 \wedge x \geq 0\}$
    $c := c + 1$
    $b := b + 1$
  $a := a - 1$
$\{B\} = \{c = \frac{(x+1)*x}{2}\}$

Weisen Sie $\models \{A\}\ p\ \{B\}$ nach, indem Sie $\models vc(\{A\}\ p\ \{B\})$ zeigen.

Abgabe bis 23.11.2020 um 10:00 Uhr per Mail an sebastian.wolff@tu-bs.de.