

Einzelprüfung „Datenbanksysteme / Softwaretechnologie (vertieft)“

Einzelprüfungsnummer 66116 / 2017 / Frühjahr

Thema 2 / Teilaufgabe 2 / Aufgabe 4 *(wp-Kalkül mit Invariante bei Methode „mul()“)*

Stichwörter: wp-Kalkül, Invariante, Totale Korrektheit

Sie dürfen im Folgenden davon ausgehen, dass keinerlei Under- oder Overflows auftreten.

Gegeben sei die folgende Methode mit Vorbedingung $P := x \geq 0 \wedge y \geq 0$ und Nachbedingung $Q := x \cdot y = z$.

```
int mul (int x , int y) {
    /* P */
    int z = 0, i = 0;
    while (i++ != x)
        z += y;
    /* Q */
    return z;
}
```

Betrachten Sie dazu die folgenden drei Prädikate:

- $I_1 := z + i \cdot y = x \cdot y$
- $I_2 := \text{false}$
- $I_3 := z + (x - i) \cdot y = x \cdot y$

- (a) Beweisen Sie formal für jedes der drei Prädikate, ob es unmittelbar vor Betreten der Schleife in `mul` gilt oder nicht.

Lösungsvorschlag

$$\begin{aligned} \text{wp}(\text{"Code vor der Schleife"}, I_1) &\equiv \text{wp}(\text{"int z = 0, i = 0;"}, z + i \cdot y = x \cdot y) \\ &\equiv \text{wp}(\text{"", } 0 + 0 \cdot y = x \cdot y) \\ &\equiv 0 = x \cdot y \\ &\equiv \text{falsch} \end{aligned}$$

$$\begin{aligned} \text{wp}(\text{"Code vor der Schleife"}, I_2) &\equiv \text{wp}(\text{"int z = 0, i = 0;"}, \text{false}) \\ &\equiv \text{wp}(\text{"", false}) \\ &\equiv \text{false} \\ &\equiv \text{falsch} \end{aligned}$$

$$\begin{aligned}
\text{wp}(\text{"Code vor der Schleife"}, I_3) &\equiv \text{wp}(\text{"int } z = 0, i = 0;", z + (x - i) \cdot y = x \cdot y) \\
&\equiv \text{wp}("", 0 + (x - 0) \cdot y = x \cdot y) \\
&\equiv x \cdot y = x \cdot y \\
&\equiv \text{wahr}
\end{aligned}$$

- (b) Weisen Sie formal nach, welche der drei Prädikate Invarianten des Schleifenrumpfs in `mul` sind oder welche nicht.

Lösungsvorschlag

Für den Nachweis muss der Code etwas umformuliert werden:

```

int mul (int x , int y) {
    /* P */
    int z = 0, i = 0;
    while (i != x) {
        i = i + 1;
        z = z + y;
    }
    /* Q */
    return z;
}

```

$$\begin{aligned}
\text{wp}(\text{"Code Schleife"}, I_1 \wedge i \neq x) &\equiv \text{wp}("i = i + 1; z = z + y;", z + i \cdot y = x \cdot y \wedge i \neq x) \\
&\equiv \text{wp}("i = i + 1;", z + y + i \cdot y = x \cdot y \wedge i \neq x) \\
&\equiv \text{wp}("", z + y + (i + 1) \cdot y = x \cdot y \wedge i + 1 \neq x) \\
&\equiv z + y + (i + 1) \cdot y = x \cdot y \wedge i + 1 \neq x \\
&\equiv z + i \cdot y + 2 \cdot y = x \cdot y \wedge i + 1 \neq x \\
&\equiv \text{falsch} \wedge i + 1 \neq x \\
&\equiv \text{falsch}
\end{aligned}$$

$$\begin{aligned}
\text{wp}(\text{"Code Schleife"}, I_2 \wedge i \neq x) &\equiv \text{wp}("i = i + 1; z = z + y;", \text{false} \wedge i \neq x) \\
&\equiv \text{wp}("", \text{false} \wedge i \neq x) \\
&\equiv \text{falsch} \wedge i \neq x \\
&\equiv \text{falsch}
\end{aligned}$$

$$\begin{aligned}
\text{wp}(\text{"Code Schleife"}, I_3 \wedge i \neq x) &\equiv \text{wp}("i = i + 1; z = z + y;", z + (x - i) \cdot y = x \cdot y \wedge i \neq x) \\
&\equiv \text{wp}("i = i + 1;", z + y + (x - i) \cdot y = x \cdot y \wedge i \neq x) \\
&\equiv \text{wp}("", z + y + (x - i + 1) \cdot y = x \cdot y \wedge i + 1 \neq x) \\
&\equiv z + y + x \cdot y - i \cdot y + y = x \cdot y \wedge i + 1 \neq x \\
&\equiv z + 2 \cdot y + x \cdot y - i \cdot y = x \cdot y \wedge i + 1 \neq x \\
&\equiv \text{wahr}
\end{aligned}$$

- (c) Beweisen Sie formal, aus welchen der drei Prädikate die Nachbedingung gefolgert werden darf bzw. nicht gefolgert werden kann.

Lösungsvorschlag

$$I_1 := z + i \cdot y = x \cdot y \quad I_2 := \text{false} \quad I_3 := z + (x - i) \cdot y = x \cdot y$$

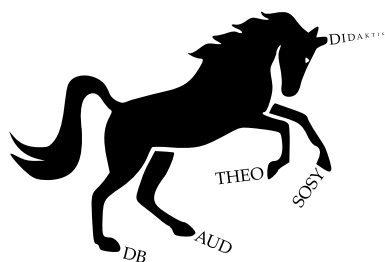
$$\begin{aligned}
\text{wp}(\text{"Code nach Schleife"}, I_1 \wedge i = x) &\equiv \text{wp}("", z + i \cdot y = x \cdot y \wedge i = x) \\
&\equiv z + i \cdot y = x \cdot y \wedge i = x \\
&\equiv z + x \cdot y = x \cdot y \\
&\neq Q
\end{aligned}$$

$$\begin{aligned}
\text{wp}(\text{"Code nach Schleife"}, I_2 \wedge i = x) &\equiv \text{wp}("", \text{false} \wedge i = x) \\
&\equiv \text{false} \wedge i = x \\
&\equiv \text{falsch} \\
&\neq Q
\end{aligned}$$

$$\begin{aligned}
\text{wp}(\text{"Code nach Schleife"}, I_3 \wedge i = x) &\equiv \text{wp}("", z + (x - i) \cdot y = x \cdot y \wedge i = x) \\
&\equiv z + (x - i) \cdot y = x \cdot y \wedge i = x \\
&\equiv z + (x - x) \cdot y = x \cdot y \\
&\equiv z + 0 \cdot y = x \cdot y \\
&\equiv z + 0 = x \cdot y \\
&\equiv z = x \cdot y \\
&\equiv Q
\end{aligned}$$

- (d) Skizzieren Sie den Beweis der totalen Korrektheit der Methode `mul`. Zeigen Sie dazu auch die Terminierung der Methode.

Aus den Teilaufgaben folgt der Beweis der partiellen Korrektheit mit Hilfe der Invariante i_3 . i steigt streng monoton von 0 an so lange gilt $i \neq x$. $i = x$ ist die Abbruchbedingung für die bedingte Wiederholung. Dann terminiert die Methode. Die Methode `mul` ist also total korrekt.



Die Bschlangaul-Sammlung

Hermine Bschlangaul and Friends

Eine freie Aufgabensammlung mit Lösungen von Studierenden für Studierende zur Vorbereitung auf die 1. Staatsexamensprüfungen des Lehramts Informatik in Bayern.



Diese Materialsammlung unterliegt den Bestimmungen der Creative Commons Namensnennung-Nicht kommerziell-Share Alike 4.0 International-Lizenz.

Hilf mit! Die Hermine schafft das nicht allein! Das ist ein Community-Projekt! Verbesserungsvorschläge, Fehlerkorrekturen, weitere Lösungen sind herzlich willkommen - egal wie - per Pull-Request oder per E-Mail an hermine.bschlangaul@gmx.net. Der \LaTeX -Quelltext dieses Dokuments kann unter folgender URL aufgerufen werden: <https://github.com/bschlangaul-sammlung/examens-aufgaben/blob/main/Staatsexamen/66116/2017/03/Thema-2/Teilaufgabe-2/Aufgabe-4.tex>