

Übungsblatt 8

Beispiellösung für Aufgabe 1

Aufgabe 1 : Beweise, wir brauchen Beweise.....

Aufgabe 1.1 ggT reloaded

Die folgende JAVA Methode soll den *grössten gemeinsamen Teiler* (ggT) zweier positiver ganzer Zahlen ermittelt.

```
public static int ggt(int wert1, int wert2)
{
    while (wert1 != wert2)
    {
        if (wert1 > wert2)
            wert1 = wert1 - wert2;
        else
            wert2 = wert2 - wert1;
    }
    return wert1;
}
```

Formal gesehen ist zu zeigen, dass dieses Programm korrekt ist bezüglich der Vorbedingung

$$PRE \equiv Z_{PRE}(wert1) > 0 \wedge Z_{PRE}(wert2) > 0$$

und der Nachbedingung

$$POST \equiv Z_{POST}(wert1) = ggT(Z_{PRE}(wert1), Z_{PRE}(wert2))$$

Für unsere Betrachtungen wollen wir den Methodenaufruf und -rückprung hier ignorieren, zu zeigen ist also

$$\{Z_{PRE}(wert1) > 0 \wedge Z_{PRE}(wert2) > 0\}$$

```
while (wert1 != wert2)
{
    if (wert1 > wert2)
        wert1 = wert1 - wert2;
    else
        wert2 = wert2 - wert1;
}
```

$$\{Z_{POST}(wert1) = ggT(Z_{PRE}(wert1), Z_{PRE}(wert2))\}$$

In Anlehnung an das Vorlesungsskript wenden wir die dort eingeführte *induktive Verifikationsstrategie* an.

Wir definieren

$$INV \equiv Z(wert1) > 0 \wedge Z(wert2) > 0 \wedge ggT(Z(wert1), Z(wert2)) = ggT(Z_{PRE}(wert1), Z_{PRE}(wert2)).$$

Wir zeigen zunächst, dass die Invariante INV vor dem Erreichen der **while**-Schleife gilt, d.h. hier im Anfangszustand Z_{PRE} .

LEMMA 1: Es gilt: $PRE \Rightarrow INV[Z_{PRE}/Z]$.

Beweis: Zu zeigen wäre

$$\begin{aligned} & (Z_{PRE}(wert1) > 0 \wedge Z_{PRE}(wert2) > 0) \Rightarrow \\ & (Z_{PRE}(wert1) > 0 \wedge Z_{PRE}(wert2) > 0 \wedge \\ & ggT(Z_{PRE}(wert1), Z_{PRE}(wert2)) = ggT(Z_{PRE}(wert1), Z_{PRE}(wert2))) \end{aligned}$$

Dieser Ausdruck ist offensichtlich wahr!

Als nächstes ist zu zeigen, dass INV tatsächlich eine Invariante für die **while**-Schleife ist, d.h. dass, wenn INV bei Eintritt in den **while**-Block im aktuellen Zustand Z gilt, INV auch am Ende des **while**-Blocks im dann erreichten Zustand Z' gilt.

LEMMA 2: Gilt INV bei Eintritt in den **while**-Block mit Valuation Z gilt, so gilt INV auch nach Ausführung des **while**-Blocks in der dann vorliegenden Valuation Z' .

Beweis: Wir berechnen, wie sich Zustand Z' aus Z ergibt durch Anwendung der semantischen Regel für das **if**-Statement.

$$Z' = \llbracket \text{if } (wert1 > wert2) \text{ wert1} = wert1 - wert2; \text{ else } wert2 = wert2 - wert1; \rrbracket (Z)$$

Aufgelöst führt dies zu

$$Z' = \begin{cases} \llbracket wert1 = wert1 - wert2; \rrbracket (Z) & \text{falls } Z(wert1) > Z(wert2) \\ \llbracket wert2 = wert2 - wert1; \rrbracket (Z) & \text{falls } Z(wert1) \leq Z(wert2) \end{cases}$$

Jetzt wird gezeigt, dass $INV[Z'/Z]$ gilt, mit der gleichen Fallunterscheidung.

Fall 1: Sei $Z(wert1) > Z(wert2)$.

Dann gilt $Z' = Z \oplus \{wert1 \mapsto Z(wert1) - Z(wert2)\}$.

- Insbesondere gilt $Z'(wert2) = Z(wert2)$ und somit $Z'(wert2) > 0$.
- Aus $Z(wert1) > 0 \wedge Z(wert2) > 0$ und $Z(wert1) > Z(wert2)$ folgt, dass $Z(wert1) - Z(wert2) > 0$, also $Z'(wert1) > 0$.
- $ggT(Z'(wert1), Z'(wert2)) = ggT(Z(wert1) - Z(wert2), Z(wert2))$,
 $ggT(Z(wert1) - Z(wert2), Z(wert2)) = ggT(Z(wert2), Z(wert1) - Z(wert2))$ und
 $Z(wert1) > Z(wert2)$, also mit den Euklidischen Axiomen
 $ggT(Z'(wert1), Z'(wert2)) = ggT(Z(wert1), Z(wert2))$
 $= ggT(Z_{PRE}(wert1), Z_{PRE}(wert2)).$

Fall 2: Sei $Z(\text{wert1}) \leq Z(\text{wert2})$. Dann gilt tatsächlich $Z(\text{wert1}) < Z(\text{wert2})$, da die Schleifenbedingung $Z(\text{wert1}) \neq Z(\text{wert2})$ für Z gilt. Also gilt $Z' = Z \oplus \{\text{wert2} \mapsto Z(\text{wert2}) - Z(\text{wert1})\}$.

- Insbesondere gilt $Z'(\text{wert1}) = Z(\text{wert1})$ und somit $Z'(\text{wert1}) > 0$.
- Aus $Z(\text{wert1}) > 0 \wedge Z(\text{wert2}) > 0$ und $Z(\text{wert2}) > Z(\text{wert1})$ folgt, dass $Z(\text{wert2}) - Z(\text{wert1}) > 0$, also $Z'(\text{wert2}) > 0$.
- $ggT(Z'(\text{wert1}), Z'(\text{wert2})) = ggT(Z(\text{wert1}), Z(\text{wert2}) - Z(\text{wert1}))$ und $Z(\text{wert1}) < Z(\text{wert2})$, also mit den Euklidischen Axiomen
 $ggT(Z'(\text{wert1}), Z'(\text{wert2})) = ggT(Z(\text{wert1}), Z(\text{wert2}))$
 $= ggT(Z_{PRE}(\text{wert1}), Z_{PRE}(\text{wert2})).$

Damit bleibt nur noch zu zeigen:

LEMMA 3: Aus Gültigkeit der Invariante und aus der Schleifenbedingung folgt, dass die Nachbedingung **POST** eingehalten wird, sobald die Schleife terminiert.

$$(INV[Z_{POST}/Z] \wedge \neg(Z_{POST}(\text{wert1}) \neq Z_{POST}(\text{wert2}))) \Rightarrow POST$$

Beweis: $\neg(Z_{POST}(\text{wert1}) \neq Z_{POST}(\text{wert2}))$ bedeutet natürlich $Z_{POST}(\text{wert1}) = Z_{POST}(\text{wert2})$. Aus $INV[Z_{POST}/Z]$ erhalten wir
 $ggT(Z_{POST}(\text{wert1}), Z_{POST}(\text{wert2})) = ggT(Z_{PRE}(\text{wert1}), Z_{PRE}(\text{wert2}))$ und wegen
 $Z_{POST}(\text{wert1}) = Z_{POST}(\text{wert2})$ und dem Euklidischen Axiom
 $Z_{POST}(\text{wert1}) = ggT(Z_{PRE}(\text{wert1}), Z_{PRE}(\text{wert2})).$

Aufgabe 1.2 Ein paar Punkte vom Weihnachtsjan

Zeigt, dass die Methode terminiert, falls **wert1** und **wert2** positive ganze Zahlen sind!

Erweiterte Strategie: Wir müssen die Strategie aus dem Skript etwas erweitern.. Das Strategieprinzip bei gegebener while-Schleife

```

1   while ( b(x_1, ..., x_n) ) {
2       B
3   }
```

lautet

- Finde eine Abbildung $f : \Sigma \mapsto W$, mit $(W, <)$ eine Wohlordnung (Σ bezeichne hier die Menge aller Zustände).

1. Finde also ein solches f , so dass für jeden beliebigen Zustand $Z \in \Sigma$ und eine Konstante $c \in W$ gilt:

$$f(Z) < c \Rightarrow \neg b(Z(x_1), \dots, Z(x_n))$$

2. Zeige, dass wenn Z den Zustand bei Eintritt in den **while**-Block (also vor **B**) bezeichnet, für den Zustand Z' nach Ausführung von **B** gilt:

$$f(Z') < f(Z)$$

Für diese Aufgabe können wir $f : \Sigma \mapsto \mathbb{N}$ nehmen, denn $\mathbb{N}$ ist Wohlordnung. Für den ggT-Algorithmus ist eine geeignete Funktion etwa $f(Z) = Z(\text{wert1}) + Z(\text{wert2})$.

Wählen wir $f(Z) = Z(\text{wert1}) + Z(\text{wert2})$. Für $c = 3$ gilt $f(Z) < c \Rightarrow Z(\text{wert1}) = Z(\text{wert2})$, da $Z(\text{wert1}) > 0$ und $Z(\text{wert2}) > 0$ aus der Invariante folgt.

Ebenso gilt wegen $Z(\text{wert1}) > 0 \wedge Z(\text{wert2}) > 0$ bei Eintritt in den **while**-Block, dass anschließend $Z'(\text{wert1}) > 0 \wedge Z'(\text{wert2}) > 0$ gilt (siehe oben).

Aus der Fallunterscheidung zur Berechnung von Z' folgt darüberhinaus, dass entweder $Z'(\text{wert1}) < Z(\text{wert1})$ gilt oder $Z'(\text{wert2}) < Z(\text{wert2})$.

In jedem Fall ist $Z'(\text{wert1}) + Z'(\text{wert2}) < Z(\text{wert1}) + Z(\text{wert2})$, d.h. $f(Z') < f(Z)$.