

Programmierung

12. Übungsblatt

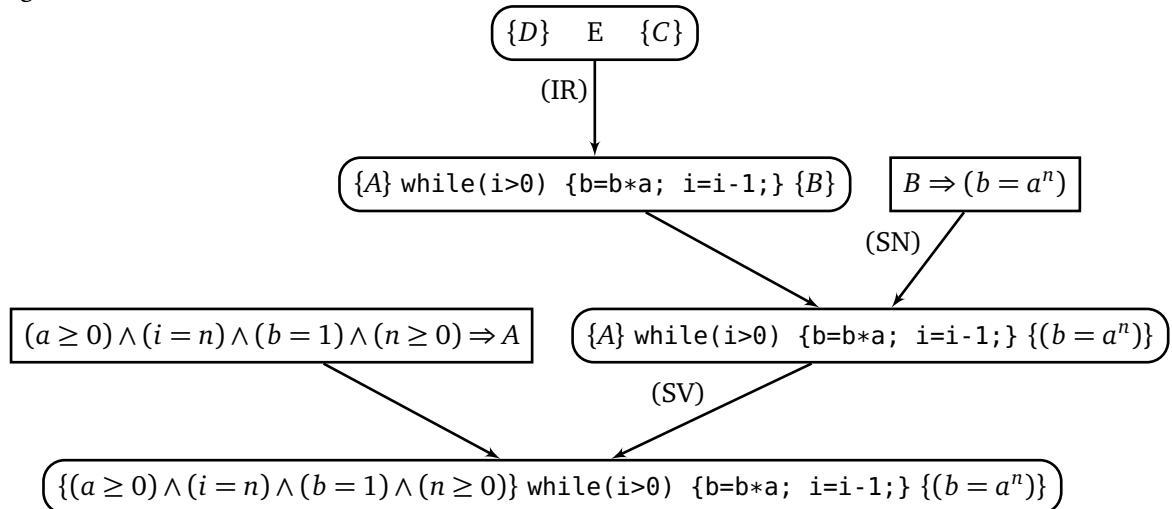
Zeitraum: 13. – 17. Juli 2015

Übung 1 (AGS 15.22)

Für die Verifikationsformel

$$\{(a \geq 0) \wedge (i = n) \wedge (b = 1) \wedge (n \geq 0)\} \text{ while } (i > 0) \{b = b * a; i = i - 1;\} \{(b = a^n)\}$$

wurden mit dem Hoare-Kalkül die ersten drei (korrekten) Regelanwendungen des Beweisbaums aufgeschrieben. Dabei sind die Ausdrücke A bis E noch unbekannt.



- Geben Sie eine geeignete Schleifeninvariante an.
- Geben Sie die Ausdrücke A , B , C , D , und E an. Sie können dabei die Schleifeninvariante mit SI abkürzen.

Übung 2 (AGS 16.4)

- Gegeben ist folgendes H_0 -Programm:

```

module Main where

test :: Int -> Int -> Int -> Int
test x1 x2 x3 = if x1 == 0 then x3
               else test (x1 - 1) x3 ((x2 * x3) + x3)

main = do x1 <- readLn
          x2 <- readLn
          print (test x2 3 x1)
    
```

Transformieren Sie dieses Programm nach den Transformationsvorschriften, wie sie in der Vorlesung angegeben wurden, in ein AM_0 -Programm mit baumstrukturierten Adressen. Sie brauchen dabei keine Zwischenschritte anzugeben.

(b) Eine Folge e_i ($i \geq 1$) von Zahlen sei wie folgt definiert:

- Die ersten beiden Glieder der Folge seien 1.
- Ab dem dritten Glied der Folge soll gelten: Jedes Folgenglied ist gleich der Summe der Quadrate der beiden Vorgängerglieder.

Geben Sie ein H_0 -Programm P an, welches das n -te Folgeelement dieser Folge berechnet und ausgibt.

Übung 3 (AGS 16.10)

Durch Nutzung der entsprechenden Transformationsfunktionen soll aus dem C_0 -Programm

```
#include <stdio.h>

int main() {
    int x1;
    scanf("%i", &x1);
    if (x1 > 0)
        while (x1 > 0) x1 = x1 - 1;
    printf("%d", x1);
    return 0;
}
```

das folgende H_0 -Programm (ohne Angabe der Funktionstypen) entstehen.

```
module Main where

f1  x1 = if x1 > 0 then A
      else B
f11 x1 = if x1 > 0 then C
      else D
f111 x1 = E
f2  x1 = F

main = do x1 <- readLn
         print (f1 x1)
```

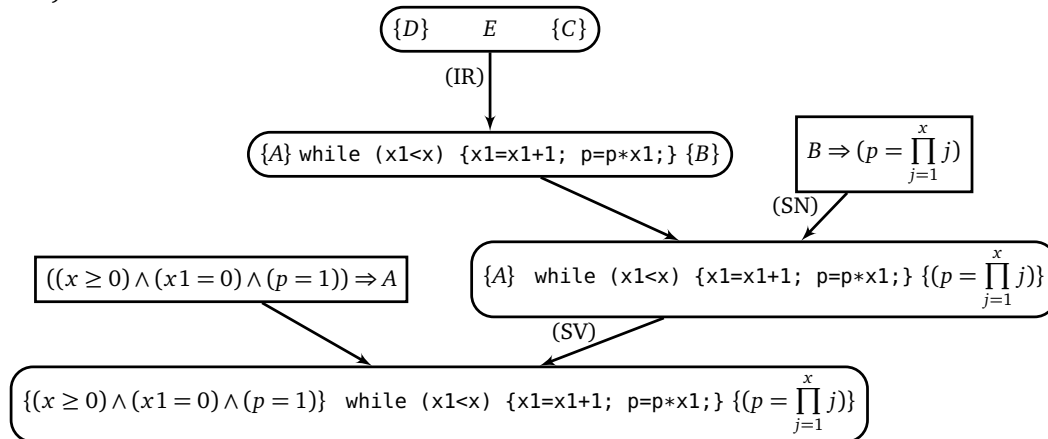
Vervollständigen Sie diesen Programmrumpf durch die Angabe der Ausdrücke für A, B, C, D, E, F!

Zusatzaufgabe 1 (AGS 15.6 ★)

Für die Verifikationsformel

$$\{(x \geq 0) \wedge (x1 = 0) \wedge (p = 1)\} \text{while } (x1 < x) \{x1 = x1 + 1; p = p * x1;\} \{(p = \prod_{j=1}^x j)\}$$

wurden die ersten drei (korrekten) Regelnwendungen des Beweisbaums aufgeschrieben (siehe unten). Dabei sind die Ausdrücke A bis E noch unbekannt.



- Geben Sie die Schleifeninvariante SI an.
- Geben Sie die Ausdrücke A , B , C , D und E an.

Zusatzaufgabe 2 (AGS 16.3 ★)

Eine Folge e_i ($i \geq 1$) von Zahlen soll wie folgt konstruiert werden:

- Das erste Glied der Folge sei 2.
- Das zweite Glied soll anwenderspezifisch gewählt werden können.
- Ab dem dritten Glied der Folge soll gelten: Jedes Folgeglied ist gleich dem Produkt der Quadrate der beiden Vorgängerglieder.

- Geben Sie eine tail-rekursive Funktion prodquad an, die das n -te Folgeelement erzeugt.
- Geben Sie ein H_0 -Programm P an, welches das n -te Folgeelement einer anwenderspezifischen Folge ausgibt.
- Übersetzen Sie das Programm P in ein entsprechendes baumstrukturiertes AM_0 -Programm. Zwischenschritte brauchen Sie nicht anzugeben.