TECHNISCHE UNIVERSITÄT MÜNCHEN
FAKULTÄT FÜR INFORMATIK

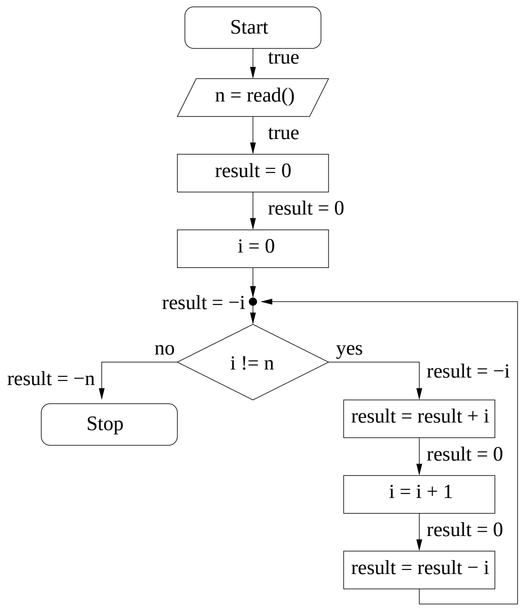
Lehrstuhl für Sprachen und Beschreibungsstrukturen
A. Berlea, F. Forster, T. Gawlitza

SoS 2007
Lösungsvorschläge zu Blatt 2

Ausgegeben am: 27. April 2007
Abgabe bis: **04. Mai 2007**

Übungen zu Einführung in die Informatik II

Aufgabe 2 Verifikation (Lösungsvorschlag)

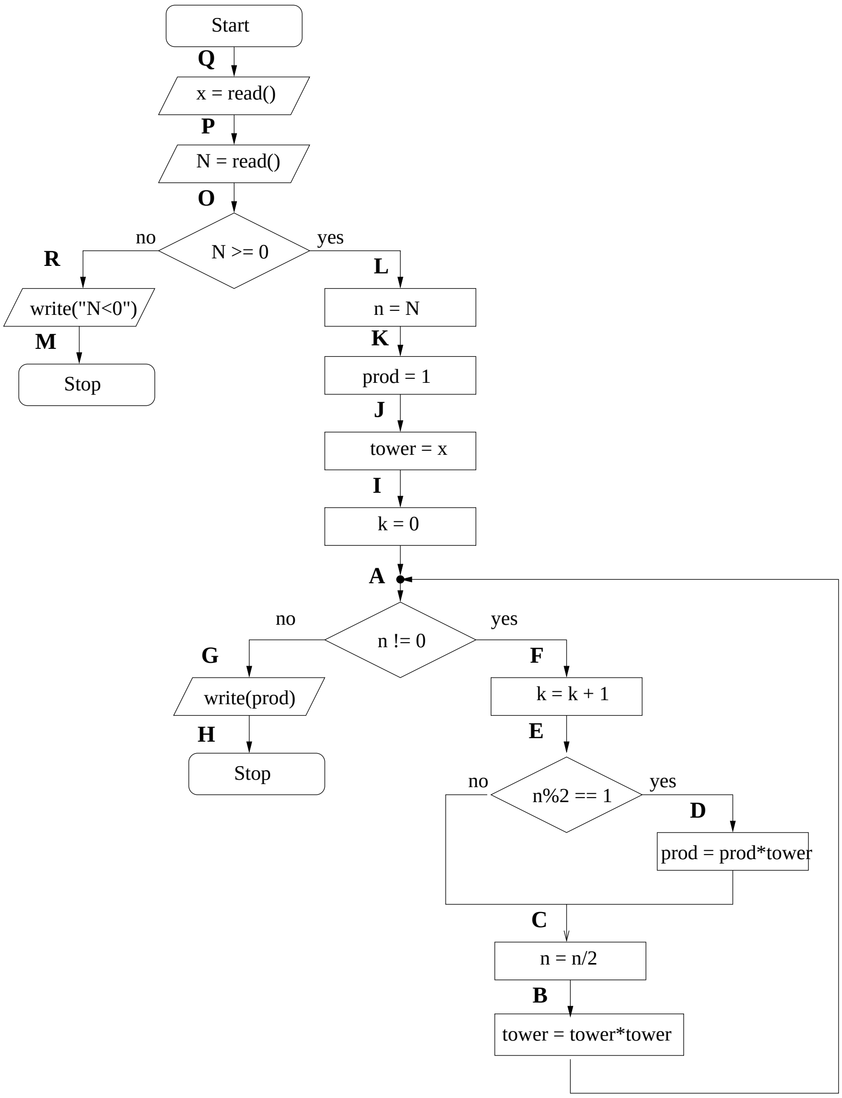


Dabei ist die einzige Stelle an der die lokale Konsistenz nicht-trivial ist der Bedingungs-Knoten. Die *weakest pre-condition* ergibt sich dort wie folgt:

$$\begin{aligned}\mathbf{WP}[\![\mathtt{i!=n}]\!](result = -n, result = -i) &\equiv (i = n \wedge result = -n) \vee (i \neq n \wedge result = -i)\\ &\equiv (i = n \wedge result = -i) \vee (i \neq n \wedge result = -i)\\ &\equiv (i = n \vee i \neq n) \wedge result = -i\\ &\equiv result = -i\end{aligned}$$

Aufgabe 3 Verifikation effizienter Berechnung von x^N (Lösungsvorschlag)

a) Der Kontrollfluss-Graph sieht wie folgt aus:



b) Die Variable `tower` wird in jedem Schleifendurchlauf quadriert. Es folgt:

$$\texttt{tower} = \texttt{x}^{2^{\texttt{k}}}$$

Die Variable `n` wird in jedem Schleifendurchlauf halbiert. Es folgt:

$$\texttt{n} = \texttt{N} \ \textbf{div}\ 2^{\texttt{k}}$$

Ensprechend dem Algorithmus hat die Variable `prod` den Wert $2^{\{b_k \ldots b_2 b_1\}_2}$, wobei $\{b_k \ldots b_2 b_1\}_2$ die letzen k binären Zahlen der Repräsentation der Zahl N zur Basis 2 darstellt. Es folgt:

$$\texttt{prod} = \texttt{x}^{\texttt{N}}\ \textbf{mod}\ 2^k$$

Die Richtigkeit dieser Gleichungen kann mit Hilfe von `assert`-Anweisungen wie im folgenden Programm übeprüft werden:

```
class Test{
    static int power(int x, int n){
        int prod=1;
```

```
        for(int i=0;i<n;i++) prod=prod*x;
        return prod;
    }

    public static void main(String[] a){
        int x = 3;
        int N = 7;

        if (N<0) {
            System.out.println("N has to be positive!");
            return;
        }

        int n=N, prod=1, tower=x, k=0;
        while(n>0){
            k=k+1;
            if (n%2==1) prod=prod*tower;
            n = n/2;
            tower = tower*tower;

            assert prod==power(x,N%power(2,k));
            assert tower==power(x,power(2,k));
            assert n==(N/power(2,k));

        }

        assert prod==power(x,N);

        System.out.println(prod);
    }
}
```

c)

- Gemäß der Überlegungen vom Punkt b) wählen wir folgende Zusicherung, die sowohl vor der Schleife als auch nach jedem Schleifendurchlauf gilt (*Schleifeninvariante*):

$$A \equiv (\texttt{tower} = \texttt{x}^{2^{\texttt{k}}}) \wedge (\texttt{n} = \texttt{N}\ \textbf{div}\ 2^{\texttt{k}}) \wedge (\texttt{prod} = \texttt{x}^{\texttt{N}\ \textbf{mod}\ 2^{k}}) \wedge (N >= 0)$$

> **Hinweis:** Alternativ kann man folgende Schleifeninvariante wählen: $\texttt{prod} = \frac{\texttt{x}^{\texttt{N}}}{\texttt{t}^{\texttt{n}}} \wedge \texttt{n} \geq 0$

- Um die lokale Konsistenz am Zuweisung-Knoten `tower=tower*tower` zu gewährleisten, wählt man für B die *weakest pre-condition* für die Zuweisung und A:

$$B \equiv \textbf{WP}[\![\texttt{tower=tower*tower}]\!](A)$$
$$\equiv (\texttt{tower*tower} = \texttt{x}^{2^{\texttt{k}}}) \wedge (\texttt{n} = \texttt{N}\ \textbf{div}\ 2^{\texttt{k}}) \wedge (\texttt{prod} = \texttt{x}^{\texttt{N}\ \textbf{mod}\ 2^{k}}) \wedge (N >= 0)$$

- Analog erhalten wir am Zuweisung-Knoten `n=n/2`:

$$C \equiv \textbf{WP}[\![\texttt{n=n/2}]\!](B)$$
$$\equiv (\texttt{tower*tower} = \texttt{x}^{2^{\texttt{k}}}) \wedge (\texttt{n}\ \textbf{div}\ 2 = \texttt{N}\ \textbf{div}\ 2^{\texttt{k}}) \wedge (\texttt{prod} = \texttt{x}^{\texttt{N}\ \textbf{mod}\ 2^{k}}) \wedge (N >= 0)$$

- Analog erhalten wir am Zuweisung-Knoten `prod=prod*tower`:

$$D \equiv \textbf{WP}[\![\texttt{prod=prod*tower}]\!](C)$$
$$\equiv (\texttt{tower*tower} = \texttt{x}^{2^{\texttt{k}}}) \wedge (\texttt{n}\ \textbf{div}\ 2 = \texttt{N}\ \textbf{div}\ 2^{\texttt{k}}) \wedge (\texttt{prod*tower} = \texttt{x}^{\texttt{N}\ \textbf{mod}\ 2^{k}}) \wedge (N >= 0)$$

- Um die lokale Konsistenz am `if(n%2==1)`-Knoten zu gewährleisten, wählt man für E die *weakest pre-condition* für die Bedingung n **mod** 2==1, die Zusicherung D an der `yes`-Kante und die Zusicherung C an der `no`-Kante:

$$
\begin{aligned}
E \equiv\ & \mathbf{WP}[\![\texttt{n\%2==1}]\!](C,D) \\
\equiv\ & (\texttt{n}\ \mathbf{mod}\ \texttt{2=0} \wedge C) \vee (\texttt{n}\ \mathbf{mod}\ \texttt{2=1} \wedge D) \\
\equiv\ & ((\texttt{n}\ \mathbf{mod}\ \texttt{2=0}) \wedge (\texttt{tower*tower} = \texttt{x}^{2^{\texttt{k}}}) \\
& \qquad \wedge (\texttt{n}\ \mathbf{div}\ 2 = \texttt{N}\ \mathbf{div}\ 2^{\texttt{k}}) \wedge (\texttt{prod} = \texttt{x}^{\texttt{N}}\ \mathbf{mod}\ 2^{k}) \wedge (N >= 0)) \\
& \vee (\texttt{n}\ \mathbf{mod}\ \texttt{2=1} \wedge (\texttt{tower*tower} = \texttt{x}^{2^{\texttt{k}}}) \wedge (\texttt{n}\ \mathbf{div}\ 2 = \texttt{N}\ \mathbf{div}\ 2^{\texttt{k}}) \\
& \qquad \wedge (\texttt{prod*tower} = \texttt{x}^{\texttt{N}}\ \mathbf{mod}\ 2^{k}) \wedge (N >= 0)) \\
\equiv\ & ((\texttt{tower*tower} = \texttt{x}^{2^{\texttt{k}}}) \wedge (\texttt{n}\ \mathbf{div}\ 2 = \texttt{N}\ \mathbf{div}\ 2^{\texttt{k}}) \wedge (N >= 0))) \\
& \wedge ((\texttt{n}\ \mathbf{mod}\ \texttt{2=0}) \wedge (\texttt{prod} = \texttt{x}^{\texttt{N}}\ \mathbf{mod}\ 2^{k}) \vee (\texttt{n}\ \mathbf{mod}\ \texttt{2=1}) \wedge (\texttt{prod*tower} = \texttt{x}^{\texttt{N}}\ \mathbf{mod}\ 2^{k}))
\end{aligned}
$$

- Um die lokale Konsistenz am Zuweisung-Knoten `k=k+1` zu gewährleisten, wählt man für F die *weakest pre-condition* für die Zuweisung und E:

$$
\begin{aligned}
F \equiv\ & \mathbf{WP}[\![\texttt{k=k+1}]\!](E) \\
\equiv\ & ((\texttt{tower*tower} = \texttt{x}^{2^{\texttt{k+1}}}) \wedge (\texttt{n}\ \mathbf{div}\ 2 = \texttt{N}\ \mathbf{div}\ 2^{\texttt{k+1}}) \wedge (N >= 0))) \wedge \\
& \qquad ((\texttt{n}\ \mathbf{mod}\ \texttt{2=0}) \wedge (\texttt{prod} = \texttt{x}^{\texttt{N}}\ \mathbf{mod}\ 2^{k+1}) \vee \\
& \qquad (\texttt{n}\ \mathbf{mod}\ \texttt{2=1}) \wedge (\texttt{prod*tower} = \texttt{x}^{\texttt{N}}\ \mathbf{mod}\ 2^{k+1}))
\end{aligned}
$$

- Um sicherzustellen, dass am Ende des Programms `prod=x`$^{\texttt{N}}$ für `N>=0` gilt, wählen wir für H:

$$H \equiv (\mathrm{N}{>=}0) \wedge (\mathrm{prod} = \mathrm{x}^{\mathrm{N}})$$

- Die Konsistenz am `write(prod)`-Knoten ist trivialerweise gewährt durch:

$$G \equiv (\mathrm{N}{>=}0) \wedge (\mathrm{prod} = \mathrm{x}^{\mathrm{N}})$$

- Um die lokale Konsistenz am `if (n!=0)`-Knoten zu überprüfen müssen wir jetzt zeigen, dass:

$$A \Rightarrow \mathbf{WP}[\![\texttt{n!=0}]\!](G,F)$$

  Wir zeigen dies, indem wir separat zeigen, dass:

  (a) $A \wedge (n = 0) \Rightarrow G$, UND

  (b) $A \wedge (n! = 0) \Rightarrow F$

  Wir beginnen mit (a). Aus $n = 0$ und $n = N$ **div** 2^k folgt, dass $N < 2^k$. Daraus folgt, dass N **mod** $2^k = N$ und schließlich, dass $prod = x^N$.
  Jetzt zeigen wir (b). Zu zeigen ist:

$$(\texttt{tower} = \texttt{x}^{2^{\texttt{k}}}) \wedge (\texttt{n} = \texttt{N}\ \mathbf{div}\ 2^{\texttt{k}}) \wedge (\texttt{prod} = \texttt{x}^{\texttt{N}}\ \mathbf{mod}\ 2^{k}) \wedge (n! = 0) \wedge (\mathrm{N}{>=}0)$$

$$\Rightarrow$$

$$
\begin{aligned}
& ((\texttt{tower*tower} = \texttt{x}^{2^{\texttt{k+1}}}) \wedge (\texttt{n}\ \mathbf{div}\ 2 = \texttt{N}\ \mathbf{div}\ 2^{\texttt{k+1}})) \\
& \wedge ((\texttt{n}\ \mathbf{mod}\ \texttt{2=0}) \wedge (\texttt{prod} = \texttt{x}^{\texttt{N}}\ \mathbf{mod}\ 2^{k+1}) \vee (\texttt{n}\ \mathbf{mod}\ \texttt{2=1}) \wedge (\texttt{prod*tower} = \texttt{x}^{\texttt{N}}\ \mathbf{mod}\ 2^{k+1}))
\end{aligned}
$$

  Wir nehmen also an, dass

$$(\texttt{tower} = \texttt{x}^{2^{\texttt{k}}}) \wedge (\texttt{n} = \texttt{N}\ \mathbf{div}\ 2^{\texttt{k}}) \wedge (\texttt{prod} = \texttt{x}^{\texttt{N}}\ \mathbf{mod}\ 2^{k}) \wedge (n! = 0) \wedge (\mathrm{N}{>=}0)$$

  gilt. Nun müssen wir zeigen, dass folgende Aussagen gelten:

(i) $((\texttt{tower*tower} = \texttt{x}^{2^{\texttt{k+1}}})$

(ii) $(\texttt{n} \ \ \mathbf{div}\ 2 = \texttt{N} \ \ \mathbf{div}\ 2^{\texttt{k+1}}))$

(iii) $((\texttt{n}\ \mathbf{mod}\ \texttt{2=0}) \wedge (\texttt{prod} = \texttt{x}^{\mathrm{N}\ \ \mathbf{mod}\ 2^{k+1}}) \vee (\texttt{n}\ \mathbf{mod}\ \texttt{2=1}) \wedge (\texttt{prod*tower} = \texttt{x}^{\mathrm{N}\ \ \mathbf{mod}\ 2^{k+1}}))$

Aussage (i) und (ii) sind offensichtlich. Um Aussage (iii) zu beweisen müssen wir zeigen, dass

(α) $\texttt{n}\ \mathbf{mod}\ \texttt{2=0} \Rightarrow \mathrm{N}\ \ \mathbf{mod}\ 2^{k+1} = \mathrm{N}\ \ \mathbf{mod}\ 2^{k}$, UND

(β) $\texttt{n}\ \mathbf{mod}\ \texttt{2=1} \Rightarrow \mathrm{N}\ \ \mathbf{mod}\ 2^{k+1} = \mathrm{N}\ \ \mathbf{mod}\ 2^{k} + 2^{k}$

Da $(\texttt{n} = \texttt{N} \ \ \mathbf{div}\ 2^{\texttt{k}})$ und $\texttt{N>=0}$, haben wir, dass:

$$\begin{aligned}
\texttt{N} &= (\texttt{N} \ \ \mathbf{div}\ 2^{\texttt{k}}) \cdot 2^{k} + \mathrm{N}\ \ \mathbf{mod}\ 2^{k} \\
&= (((\texttt{N} \ \ \mathbf{div}\ 2^{\texttt{k}})\ \mathbf{div}\ 2) \cdot 2 + (\texttt{N} \ \ \mathbf{div}\ 2^{\texttt{k}})\ \mathbf{mod}\ 2) \cdot 2^{k} + \mathrm{N}\ \ \mathbf{mod}\ 2^{k} \\
&= (((\texttt{N} \ \ \mathbf{div}\ 2^{\texttt{k}})\ \mathbf{div}\ 2) \cdot 2 + n\ \mathbf{mod}\ 2) \cdot 2^{k} + \mathrm{N}\ \ \mathbf{mod}\ 2^{k} \\
&= (\texttt{N} \ \ \mathbf{div}\ 2^{\texttt{k+1}}) \cdot 2^{k+1} + 2^{k} \cdot (n\ \mathbf{mod}\ 2) + \mathrm{N}\ \ \mathbf{mod}\ 2^{k}
\end{aligned}$$

Außerdem gilt, dass:

$$\texttt{N} = (\texttt{N} \ \ \mathbf{div}\ 2^{\texttt{k+1}}) \cdot 2^{k+1} + \mathrm{N}\ \ \mathbf{mod}\ 2^{k+1}$$

Aus den letzen zwei Gleichheiten folgt, dass:

$$\mathrm{N}\ \ \mathbf{mod}\ 2^{k+1} = \mathrm{N}\ \ \mathbf{mod}\ 2^{k} + 2^{k} \cdot (n\ \mathbf{mod}\ 2)$$

Daraus folgen direkt (α) und (β).

- Um die lokale Konsistenz am Zuweisung-Knoten `k=0` zu gewährleisten, wählt man für I die *weakest pre-condition* für die Zuweisung und A:

$$\begin{aligned}
I &\equiv \mathbf{WP}[\![\texttt{k=0}]\!](A) \\
&\equiv (\texttt{tower} = \texttt{x}) \wedge (\texttt{n} = \texttt{N}) \wedge (\texttt{prod} = 1) \wedge (N >= 0)
\end{aligned}$$

- Um die lokale Konsistenz am Zuweisung-Knoten `tower=x` zu gewährleisten, wählt man für J die *weakest pre-condition* für die Zuweisung und I:

$$\begin{aligned}
J &\equiv \mathbf{WP}[\![\texttt{tower=x}]\!](I) \\
&\equiv (\texttt{n} = \texttt{N}) \wedge (\texttt{prod} = 1) \wedge (N >= 0)
\end{aligned}$$

- Um die lokale Konsistenz am Zuweisung-Knoten `prod=1` zu gewährleisten, wählt man für K die *weakest pre-condition* für die Zuweisung und J:

$$\begin{aligned}
K &\equiv \mathbf{WP}[\![\texttt{prod=1}]\!](J) \\
&\equiv (\texttt{n} = \texttt{N}) \wedge (N >= 0)
\end{aligned}$$

- Um die lokale Konsistenz am Zuweisung-Knoten `n=N` zu gewährleisten, wählt man für L die *weakest pre-condition* für die Zuweisung und K:

$$\begin{aligned}
L &\equiv \mathbf{WP}[\![\texttt{n=N}]\!](K) \\
&\equiv (N >= 0)
\end{aligned}$$

- $M \equiv R \equiv N < 0$
- $O \equiv \mathbf{WP}[\![\texttt{N>=0}]\!](R, L) \equiv \mathit{true}$
- Die lokale Konsistenz ist vollständig überprüft durch $P \equiv Q \equiv \mathit{true}$.