Aufgabe 3: wp-Kalkül und Schleifeninvariante

Gegeben Sei folgendes Programm:

```
long doubleFac (long n) {
  /* P */ long df = 1;
  for (long x = n; x > 1; x -= 2) {
    df *= x;
  } /* Q */
  return df;
}
```

sowie die Vorbedingung $P \equiv n \geq 0$ und Nachbedingung $Q \equiv (\mathrm{df} = n!!)$ wobei gilt

$$n!! := \begin{cases} 2^k \cdot k! & n \text{ gerade}, k := \frac{n}{2} \\ \frac{(2k)!}{2^k \cdot k!} & n \text{ ungerade}, k := \frac{n+1}{2} \end{cases}$$

> **Exkurs: Fakultät**
> Die Fakultät ist eine Funktion, die einer natürlichen Zahl das Produkt aller natürlichen Zahlen (ohne Null) kleiner und gleich dieser Zahl zuordnet. Für alle natürlichen Zahlen n ist
>
> $$n! = 1 \cdot 2 \cdot 3 \cdots n = \prod_{k=1}^{n} k$$

> **Exkurs: Doppelfakultät**
> Die seltener verwendete „Doppelfakultät“ oder „doppelte Fakultät“ ist für gerade n das Produkt aller geraden Zahlen kleiner gleich n. Für ungerade n ist es das Produkt aller ungeraden Zahlen kleiner gleich n. Sie ist definiert als:
>
> $$n!! = \begin{cases} n \cdot (n-2) \cdot (n-4) \cdots 2 & \text{für } n \text{ gerade und } n > 0, \\ n \cdot (n-2) \cdot (n-4) \cdots 1 & \text{für } n \text{ ungerade und } n > 0, \\ 1 & \text{für } n \in \{-1, 0\} \end{cases}$$
>
> Häufig werden anstelle der Doppelfakultät Ausdrücke mit der gewöhnlichen Fakultät verwendet. Es gilt $(2k)!! = 2^k k!$ und $(2k-1)!! = \frac{(2k)!}{2^k k!}$

Zur Vereinfachung nehmen Sie im Folgenden an, dass die verwendeten Datentypen unbeschränkt sind und daher keine Überläufe auftreten können.

(a) Welche der folgenden Bedingungen ist eine zum Beweisen der Korrektheit der Methode mittels wp-Kalkül (Floyd-Hoare-Kalkül) sinnvolle Schleifeninvariante?

  (i) $\mathrm{df} = n!! - x!! \wedge x \geq 1$
  (ii) $\mathrm{df} = (n-x)!! \wedge x \geq 1$
  (iii) $\mathrm{df} \cdot x!! = n!! \wedge x \geq 0$
  (iv) $(\mathrm{df} + x)!! = n!! \wedge x \geq 0$

Zunächst wird der Code in einen äquivalenten Code mit while-Schleife umgewandelt:

```
long doubleFac (long n) {
  /* P */ long df = 1;
  long x = n ;
  while (x > 1) {
    df = df * x;
    x = x - 2;
  } /* Q */
  return df;
}
```

(i) $\mathrm{df} = n!! - x!! \wedge x \geq 1$

(ii) $\mathrm{df} = (n - x)!! \wedge x \geq 1$

Die ersten beiden Bedingungen sind unmöglich, da z. B. für $n = 2$ nach der Schleife $x = 0$ gilt und daher $x \geq 1$ verletzt wäre.

(iii) $\mathrm{df} \cdot x!! = n!! \wedge x \geq 0$

Nach dem Ausschlussprinzip ist es daher die dritte Bedingung: $I \equiv (\mathrm{df} + x)!! = n!! \wedge x \geq 0$.

(iv) $(\mathrm{df} + x)!! = n!! \wedge x \geq 0$

Die letzte kann es auch nicht sein, da vor der Schleife $\mathrm{df} = 1$ und $x = n$ gilt, d. h. $(\mathrm{df} + x)!! = (1 + n)!!$. Jedoch ist offenbar $(1 + n)!! \neq n!!$.

$\Rightarrow$ Die Schleifeninvariante lautet: $\mathrm{df} \cdot x!! = n!! \wedge x \geq 0$

(b) Zeigen Sie formal mittels wp-Kalkül, dass die von Ihnen gewählte Bedingung unmittelbar vor Beginn der Schleife gilt, wenn zu Beginn der Methode die Anfangsbedingung P gilt.

Zu zeigen $P \Rightarrow \mathrm{wp}(\texttt{"Code vor der Schleife"}, I)$

$$\begin{aligned}
\mathrm{wp}(\texttt{"Code vor der Schleife"}, I) &\equiv \mathrm{wp}(\texttt{"df = 1; x = n;"}, (\mathrm{df} \cdot x)!! = n!! \wedge x \geq 0) \\
&\equiv \mathrm{wp}(\texttt{"df = 1;"}, (\mathrm{df} \cdot n)!! = n!! \wedge n \geq 0) \\
&\equiv \mathrm{wp}(\texttt{""}, (1 \cdot n)!! = n!! \wedge n \geq 0) \\
&\equiv n!! = n!! \wedge n \geq 0 \\
&\equiv n \geq 0 \\
&\equiv P
\end{aligned}$$

Insbesondere folgt damit die Behauptung.

(c) Zeigen Sie formal mittels wp-Kalkül, dass die von Ihnen gewählte Bedingung tatsächlich eine Invariante der Schleife ist.

zu zeigen: $I \wedge$ Schleifenbedingung $\Rightarrow$ wp(`"Code in der Schleife"`, I)

Bevor wir dies beweisen, zeigen wir erst $x \cdot (x-2)!! = x!!$.

- Fall x ist gerade ($n!! = 2^k \cdot k!$ für $k := \frac{n}{2}$):
  $x \cdot (x-2)!! = x \cdot 2^{\frac{x-2}{2}} \cdot (\frac{x-2}{2})! = x \cdot \frac{1}{2} \cdot 2^{\frac{x}{2}} \cdot (\frac{x}{2} - 1)! = 2^{\frac{x}{2}} \cdot (\frac{x}{2})! = x!!$
  Nebenrechnung (Division mit gleicher Basis: $x^{a-b} = \frac{x^a}{x^b}$):
  $2^{\frac{x-2}{2}} = 2^{(\frac{x}{2} - \frac{2}{2})} = \frac{2^{\frac{x}{2}}}{2^{\frac{2}{2}}} = \frac{2^{\frac{x}{2}}}{2^1} = \frac{2^{\frac{x}{2}}}{2} = \frac{1}{2} \cdot 2^{\frac{x}{2}}$
  Nebenrechnung ($n! = (n-1)! \cdot n$):
  $x \cdot \frac{1}{2} \cdot (\frac{x}{2} - 1)! = \frac{x}{2} \cdot (\frac{x}{2} - 1)! = \frac{x}{2}!$
- Fall x ist ungerade:

Dies benutzen wir nun, um den eigentlichen Beweis zu führen:

$$\begin{aligned}
\text{wp}(\texttt{"Code vor der Schleife"}, I) &\equiv \text{wp}(\texttt{"df = df * x; x = x - 2;"}, (\text{df} \cdot x)!! = n!! \wedge x \geq 0) \\
&\equiv \text{wp}(\texttt{"df = df * x;"}, (\text{df} \cdot (x-2)))!! = n!! \wedge x - 2 \geq 0) \\
&\equiv \text{wp}(\texttt{""}, (\text{df} \cdot x \cdot (x-2)))!! = n!! \wedge x - 2 \geq 0) \\
&\equiv (\text{df} \cdot x)!! = n!! \wedge x \geq 2 \\
&\equiv (\text{df} \cdot x)!! = n!! \wedge x > 1 \\
&\equiv I \wedge x > 1 \\
&\equiv I \wedge \text{Schleifenbedingung}
\end{aligned}$$

(d) Zeigen Sie formal mittels wp-Kalkül, dass am Ende der Methode die Nachbedingung Q erfüllt wird.

z.z. $I \wedge \neg$Schleifenbedingung $\Rightarrow$ wp(`"Code nach der Schleife"`, Q)

Wir vereinfachen den Ausdruck $I \wedge \neg$Schleifenbedingung:

$I \wedge \neg\text{Schleifenbedingung} \equiv I \wedge (x \leq 1) \equiv I \wedge ((x = 0) \vee (x = 1)) \equiv (I \wedge (x = 0)) \vee (I \wedge (x = 1)) \equiv (df \cdot 1 = n!!) \vee (df \cdot 1 = n!!) \equiv df = n!!$

Damit gilt:

$\text{wp}(\texttt{"Code nach der Schleife"}, Q) \equiv \text{wp}(\texttt{""}, df = n!!) \equiv df = n!! \equiv I \wedge \neg\text{Schleifenbedingung}$

(e) Beweisen Sie, dass die Methode immer terminiert. Geben Sie dazu eine Terminierungsfunktion an und begründen Sie kurz ihre Wahl.

Sei $T(x) := x$. T ist offenbar ganzzahlig. Da x in jedem Schleifendurchlauf um 2 verringert wird, ist T streng monoton fallend. Aus der Schleifeninvariante folgt $x \geq 0$ und daher ist x auch nach unten beschränkt. Damit folgt $I \Rightarrow T \geq 0$ und T ist eine gültige Terminierungsfunktion.