Technische Universität München
Institut für Informatik
Prof. Tobias Nipkow, Ph.D.
Dr. Olaf Müller

WS 1998/99
Lösungsvorschläge zu Blatt 4
1. 12. 1998

Übungen zur Semantik von Programmiersprachen

Aufgabe 1 (Lösungsvorschlag)

(a) In der Vorlesung wurde gezeigt, daß für alle $c \in \mathbf{Com}$ die denotationelle Semantik $\mathcal{C}[\![c]\!] \subseteq \Sigma \times \Sigma$ eine partielle Funktion ist. Deshalb können wir im folgenden statt der etwas umständlicheren Relationenschreibweise die Funktionsschreibweise verwenden.

Da die denotationelle Semantik kompositional ist, können wir allein die Semantik des Schleifenrumpfs berechnen und dieses Ergebnis später wiederverwenden.

$$
\begin{array}{ll}
 & \mathcal{C}[\![X := X + 2 \cdot i\ ;\ i := i - 1]\!]\sigma \\
= & \{\ \text{Def. } \mathcal{C} \text{ für sequentielle Komposition}\} \\
 & \mathcal{C}[\![i := i - 1]\!](\mathcal{C}[\![X := X + 2 \cdot i]\!]\sigma) \\
= & \{\ \text{Def. } \mathcal{C} \text{ für Zuweisung}\} \\
 & \mathcal{C}[\![i := i - 1]\!](\sigma[\mathcal{A}[\![X + 2 \cdot i]\!]\sigma / X]) \\
= & \{\ \text{Def. } \mathcal{A}\} \\
 & \mathcal{C}[\![i := i - 1]\!](\sigma[\sigma(X) + 2 \cdot \sigma(i) / X]) \\
= & \{\ \text{Def. } \mathcal{C} \text{ für Zuweisung}\} \\
 & (\sigma[\sigma(X) + 2 \cdot \sigma(i)/X])[\mathcal{A}[\![i - 1]\!](\sigma[\sigma(X) + 2 \cdot \sigma(i)/X])/i] \\
= & \{\ \text{Def. } \mathcal{A} \text{ und } (\sigma[\sigma(X) + 2 \cdot \sigma(i)/X])(i) = \sigma(i)\} \\
 & (\sigma[\sigma(X) + 2 \cdot \sigma(i)/X])[\sigma(i) - 1/i] \\
= & \{\ \text{simultanes Überschreiben}\} \\
 & \sigma[\sigma(X) + 2 \cdot \sigma(i)/X, \sigma(i) - 1/i]
\end{array}
$$

Damit ergibt sich für $\Gamma : \ (\Sigma \rightharpoonup \Sigma) \to (\Sigma \rightharpoonup \Sigma)$

$$
\begin{array}{ll}
 & \Gamma(\varphi) \\
= & \qquad \{\text{Definition}\} \\
 & \{(\sigma, \sigma') | \exists \sigma''.\ \ (\mathcal{B}[\![i > 0]\!])\sigma = \mathbf{true}\ \wedge (\sigma, \sigma'') \in \mathcal{C}[\![X := X + 2 \cdot i\ ;\ i := i - 1]\!] \wedge \\
 & \qquad\qquad (\sigma'', \sigma') \in \varphi\} \cup \\
 & \{(\sigma, \sigma) | (\mathcal{B}[\![i > 0]\!])\sigma = \mathbf{false}\} \\
= & \qquad \{\text{Definition von } \mathcal{B} \text{ und Ergebnis für } \mathcal{C}\} \\
 & \{(\sigma, \sigma') | \exists \sigma''.\ \ \sigma(i) > 0\ \wedge \sigma'' = \sigma[\sigma(X) + 2 \cdot \sigma(i)/X, \sigma(i) - 1/i] \wedge \\
 & \qquad\qquad (\sigma'', \sigma') \in \varphi\} \cup \\
 & \{(\sigma, \sigma) | \sigma(i) \leq 0\} \\
= & \\
 & \{(\sigma, \sigma') |\ \ \sigma(i) > 0\ \wedge (\sigma[\sigma(X) + 2 \cdot \sigma(i)/X, \sigma(i) - 1/i], \sigma') \in \varphi\} \cup \\
 & \{(\sigma, \sigma) | \sigma(i) \leq 0\}
\end{array}
$$

(b)

$$
\begin{array}{ll}
 & \Gamma^0(\emptyset) \\
= & \qquad \{\text{Definition von } \Gamma^0\}
\end{array}
$$

$$
\begin{array}{ll}
 & \emptyset \\
 & \\
 & \Gamma^1(\emptyset) \\
= & \qquad \{\text{Definition von } \Gamma^1\} \\
 & \Gamma(\Gamma^0(\emptyset)) \\
= & \qquad \{\text{Teilaufgabe (a)}\} \\
 & \{(\sigma,\sigma')|\quad \sigma(i) > 0 \ \wedge (\sigma[\sigma(X) + 2\cdot\sigma(i)/X, \sigma(i) - 1/i], \sigma') \in \Gamma^0(\emptyset)\} \cup \\
 & \{(\sigma,\sigma)|\sigma(i) \leq 0\} \\
= & \qquad \{\Gamma^0(\emptyset) = \emptyset\} \\
 & \{(\sigma,\sigma)|\sigma(i) \leq 0\} \\
 & \\
 & \Gamma^2(\emptyset) \\
= & \qquad \{\text{Definition von } \Gamma^2\} \\
 & \Gamma(\Gamma^1(\emptyset)) \\
= & \qquad \{\text{Teilaufgabe (a)}\} \\
 & \{(\sigma,\sigma')|\quad \sigma(i) > 0 \ \wedge (\sigma[\sigma(X) + 2\cdot\sigma(i)/X, \sigma(i) - 1/i], \sigma') \in \Gamma^1(\emptyset)\} \cup \\
 & \{(\sigma,\sigma)|\sigma(i) \leq 0\} \\
= & \qquad \{\Gamma^1(\emptyset) \text{ s.o.}\} \\
 & \{(\sigma,\sigma')|\quad \sigma(i) > 0 \ \wedge (\sigma[\sigma(X) + 2\cdot\sigma(i)/X, \sigma(i) - 1/i], \sigma') \in \{(\sigma,\sigma)|\sigma(i) \leq 0\}\} \cup \\
 & \{(\sigma,\sigma)|\sigma(i) \leq 0\} \\
= & \\
 & \{(\sigma, \sigma[\sigma(X) + 2\cdot\sigma(i)/X, \sigma(i) - 1/i])|\sigma(i) = 1\} \cup \\
 & \{(\sigma,\sigma)|\sigma(i) \leq 0\} \\
 & \\
 & \Gamma^3(\emptyset) \\
= & \qquad \{\text{Definition von } \Gamma^3\} \\
 & \Gamma(\Gamma^2(\emptyset)) \\
= & \qquad \{\text{Teilaufgabe (a)}\} \\
 & \{(\sigma,\sigma')|\quad \sigma(i) > 0 \ \wedge (\sigma[\sigma(X) + 2\cdot\sigma(i)/X, \sigma(i) - 1/i], \sigma') \in \Gamma^2(\emptyset)\} \cup \\
 & \{(\sigma,\sigma)|\sigma(i) \leq 0\} \\
= & \qquad \{\Gamma^2(\emptyset) \text{ s.o.}\} \\
 & \{(\sigma, \sigma[\sigma(X) + 2\cdot\sigma(i) + 2\cdot(\sigma(i) - 1)/X, \sigma(i) - 2/i])|\sigma(i) = 2\} \cup \\
 & \{(\sigma, \sigma[\sigma(X) + 2\cdot\sigma(i)/X, \sigma(i) - 1/i])|\sigma(i) = 1\} \cup \\
 & \{(\sigma,\sigma)|\sigma(i) \leq 0\} \\
= & \qquad \{\text{Arithmetik}\} \\
 & \{(\sigma, \sigma[\sigma(X) + 2\cdot\sum_{k=1}^{\sigma(i)} k/X, 0/i])|1 \leq \sigma(X) \leq 2\} \cup \\
 & \{(\sigma,\sigma)|\sigma(i) \leq 0\}
\end{array}
$$

(c) In Funktionsschreibweise können wir die denotationelle Semantik der *while*-Schleife ausdrücken als

$$\mathcal{C}[\![\textbf{while } b \textbf{ do } c]\!] = fix(\Gamma)$$

mit

$$\Gamma : (\Sigma \nrightarrow \Sigma) \to (\Sigma \nrightarrow \Sigma)$$

$$\Gamma(\varphi)(\sigma) = \begin{cases} \varphi(\mathcal{C}[\![c]\!]\sigma) & \text{falls} \quad \mathcal{B}[\![b]\!]\sigma = \mathbf{true} \\ \sigma & \text{falls} \quad \mathcal{B}[\![b]\!]\sigma = \mathbf{false}. \end{cases}$$

Insbesondere können wir zeigen, daß $\mathcal{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]$ Fixpunkt ist von Γ:

$$\begin{aligned} & \lambda\sigma. \begin{cases} \mathcal{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!](\mathcal{C}[\![c]\!]\sigma) & \text{falls} \quad \mathcal{B}[\![b]\!]\sigma = \mathbf{true} \\ \sigma & \text{falls} \quad \mathcal{B}[\![b]\!]\sigma = \mathbf{false} \end{cases} \\ = & \quad \{ \quad \text{Def. } \Gamma \} \\ & \Gamma(\mathcal{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]) \\ = & \quad \{ \quad \text{Def. von } \mathcal{C} \text{ für die } \textit{while}\text{-Schleife} \} \\ & \Gamma(\mathit{fix}(\Gamma)) \\ = & \quad \{ \quad \text{Fixpunkteigenschaft} \} \\ & \mathit{fix}(\Gamma) \\ = & \quad \{ \quad \text{Def. von } \mathcal{C} \text{ für die } \textit{while}\text{-Schleife} \} \\ & \mathcal{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!] \end{aligned}$$

Die damit bewiesene Gleichheit zwischen der ersten und der letzten Zeile nennen wir im folgenden die "Fixpunkteigenschaft von *while*".

Kürzen wir ab jetzt den Schleifenrumpf $X := X + 2 \cdot i\ ;\ i := i - 1$ mit c ab.

Wegen des Hinweises "Induktion über $\sigma(i)$" formen wir die Behauptung folgendermaßen um:

$$\forall n. \forall \sigma. \sigma(i) = n \Rightarrow (\mathcal{C}[\![\mathbf{while}\ i > 0\ \mathbf{do}\ c]\!]\sigma)(X) = \sigma(i) \cdot (\sigma(i) + 1) + \sigma(X)$$

mittels Induktion über n:

Induktionsanfang $n = 0$: Sei σ beliebig mit $\sigma(i) = 0$.

$$\begin{aligned} & (\mathcal{C}[\![\mathbf{while}\ i > 0\ \mathbf{do}\ c]\!]\sigma)(X) \\ = & \quad \{ \quad \text{Fixpunkteigenschaft von } \textit{while} \} \\ & \begin{cases} \mathcal{C}[\![\mathbf{while}\ i > 0\ \mathbf{do}\ c]\!](\mathcal{C}[\![c]\!]\sigma)(X) & \text{falls} \quad \mathcal{B}[\![i > 0]\!]\sigma = \mathbf{true} \\ \sigma(X) & \text{falls} \quad \mathcal{B}[\![i > 0]\!]\sigma = \mathbf{false} \end{cases} \\ = & \quad \{ \quad \text{nach Voraussetzung ist } \sigma(i) = 0\text{; Def. von } \mathcal{B} \text{ liefert } \mathcal{B}[\![i > 0]\!]\sigma = \mathbf{false} \} \\ & \sigma(X) \\ = & \quad \{ \quad \text{Arithmetik; } \sigma(i) = 0 \} \\ & \sigma(i) \cdot (\sigma(i) + 1) + \sigma(X) \end{aligned}$$

Induktionsschritt von n nach $n + 1$:

Induktionsvoraussetzung:

$$\forall \sigma. \sigma(i) = n \Rightarrow (\mathcal{C}[\![\mathbf{while}\ i > 0\ \mathbf{do}\ c]\!]\sigma)(X) = \sigma(i) \cdot (\sigma(i) + 1) + \sigma(X)$$

Sei σ beliebig mit $\sigma(i) = n + 1$.

$$
\begin{array}{cl}
 & (\mathcal{C}[\![\textbf{while } i > 0 \textbf{ do } c]\!]\sigma)(X) \\
= & \{\ \text{Fixpunkteigenschaft von } \textit{while}\} \\
 & \begin{cases} \mathcal{C}[\![\textbf{while } i > 0 \textbf{ do } c]\!](\mathcal{C}[\![c]\!]\sigma)(X) & \text{falls} \quad \mathcal{B}[\![i > 0]\!]\sigma = \textbf{true} \\ \sigma(X) & \text{falls} \quad \mathcal{B}[\![i > 0]\!]\sigma = \textbf{false} \end{cases} \\
= & \{\ \text{nach Voraussetzung ist } \sigma(i) = n + 1\text{; Def. von } \mathcal{B} \text{ liefert } \mathcal{B}[\![i > 0]\!]\sigma = \textbf{true}\} \\
 & \mathcal{C}[\![\textbf{while } i > 0 \textbf{ do } c]\!](\mathcal{C}[\![c]\!]\sigma)X \\
= & \{\ \text{obige Rechnung für den Schleifenrumpf}\} \\
 & \mathcal{C}[\![\textbf{while } i > 0 \textbf{ do } c]\!](\sigma[\sigma(X) + 2 \cdot \sigma(i)/X, \sigma(i) - 1/i]) \\
= & \{\ \text{Induktionsvoraussetung mit } \sigma[\sigma(X) + 2 \cdot \sigma(i)/X, \sigma(i) - 1/i] \text{ für } \sigma; \\
 & \quad \text{es gilt } (\sigma[\sigma(X) + 2 \cdot \sigma(i)/X, \sigma(i) - 1/i])(i) = n\} \\
 & (\sigma[\sigma(X) + 2 \cdot \sigma(i)/X, \sigma(i) - 1/i])(i) \cdot ((\sigma[\sigma(X) + 2 \cdot \sigma(i)/X, \sigma(i) - 1/i])(i) + 1) + \\
 & (\sigma[\sigma(X) + 2 \cdot \sigma(i)/X, \sigma(i) - 1/i])(X) \\
= & \{\ \text{Def. des Überschreibens}\} \\
 & (\sigma(i) - 1) \cdot (\sigma(i) - 1 + 1) + \sigma(X) + 2 \cdot \sigma(i) \\
= & \{\ \text{Arithmetik}\} \\
 & \sigma(i) \cdot (\sigma(i) + 1) + \sigma(X)
\end{array}
$$

(d) Für alle $n \in \mathbb{N}$ gilt:

$$
\begin{array}{cl}
 & \Gamma^n(\emptyset) \\
= & \qquad \{\text{ohne Beweis}\} \\
 & \{(\sigma, \sigma[\sigma(X) + \sigma(i) \cdot (\sigma(i) + 1)/X, 0/i]) | 1 \le \sigma(i) < n\} \cup \\
 & \{(\sigma, \sigma) | \sigma(i) \le 0 \wedge n > 0\}
\end{array}
$$

$$\bigcup_{n \in \mathbb{N}} \Gamma^n(\emptyset) = \{(\sigma, \sigma[\sigma(X) + \sigma(i) \cdot (\sigma(i) + 1)/X, 0/i])\}$$

Dieses Ergebnis konnten wir erwarten, denn es gilt

$$\mathcal{C}[\![w]\!] = fix(\Gamma) = \bigcup_{n \in \mathbb{N}} \Gamma^n(\emptyset)$$

(e) *Zu zeigen*

$$\forall \sigma. (\mathcal{C}[\![P]\!]\sigma)(X) = \sigma(N) \cdot (\sigma(N) + 1)$$

Sei σ beliebig. Wir schreiben w für die *while*-Schleife.

$$
\begin{array}{cl}
 & \mathcal{C}[\![X := 0\ ;\ i := N\ ;\ w]\!]\sigma X \\
= & \{\ \text{Def. von } \mathcal{C} \text{ für die sequentielle Komposition}\} \\
 & \mathcal{C}[\![w]\!](\mathcal{C}[\![X := 0\ ;\ i := N]\!]\sigma)X \\
= & \{\ \text{Def. von } \mathcal{C} \text{ für sequentielle Komposition und Zuweisung}\} \\
 & \mathcal{C}[\![w]\!](\sigma[0/X, \sigma(N)/i])X \\
= & \{\ \text{Teilaufgabe (a); Arithmetik}\} \\
 & \sigma(N) \cdot (\sigma(N) + 1)
\end{array}
$$

Aufgabe 2 (Lösungsvorschlag)

Wir orientieren uns an der denotationellen Semantik von **while** und entwickeln die Semantik von **repeat** in analoger Weise. Sei

$$w \equiv \textbf{repeat}\ c\ \textbf{until}\ b$$

Beobachtung bzw. Forderung:

$$w \sim c; \textbf{if}\ b\ \textbf{then skip else}\ w$$

Die partielle Funktion $\mathcal{C}[\![w]\!]$ sollte also identisch zu $\mathcal{C}[\![c; \textbf{if}\ b\ \textbf{then skip else}\ w]\!]$ sein.

$$\begin{aligned}
\mathcal{C}[\![w]\!] &= \mathcal{C}[\![c; \textbf{if}\ b\ \textbf{then skip else}\ w]\!] \\
&= \mathcal{C}[\![\textbf{if}\ b\ \textbf{then skip else}\ w]\!] \circ \mathcal{C}[\![c]\!] \\
&= (\{(\sigma',\sigma'') \mid \mathcal{B}[\![b]\!](\sigma') = \mathit{false} \wedge (\sigma',\sigma'') \in \mathcal{C}[\![w]\!]\} \cup \\
&\quad\ \{(\sigma',\sigma'') \mid \mathcal{B}[\![b]\!](\sigma') = \mathit{true} \wedge (\sigma',\sigma'') \in \mathcal{C}[\![\textbf{skip}]\!]\}) \circ \mathcal{C}[\![c]\!] \\
&= (\{(\sigma',\sigma'') \mid \mathcal{B}[\![b]\!](\sigma') = \mathit{false} \wedge (\sigma',\sigma'') \in \mathcal{C}[\![w]\!]\} \circ \mathcal{C}[\![c]\!]) \cup \\
&\quad\ (\{(\sigma',\sigma') \mid \mathcal{B}[\![b]\!](\sigma') = \mathit{true}\} \circ \mathcal{C}[\![c]\!]) \\
&= \{(\sigma,\sigma'') \mid \exists \sigma'.\ (\sigma,\sigma') \in \mathcal{C}[\![c]\!] \wedge \mathcal{B}[\![b]\!](\sigma') = \mathit{false} \wedge (\sigma',\sigma'') \in \mathcal{C}[\![w]\!]\} \cup \\
&\quad\ \{(\sigma,\sigma') \mid (\sigma,\sigma') \in \mathcal{C}[\![c]\!] \wedge \mathcal{B}[\![b]\!](\sigma') = \mathit{true}\}
\end{aligned}$$

Die entstehende Gleichung ist rekursiv in $\mathcal{C}[\![w]\!]$. Wir führen die Abkürzung $\beta \equiv \mathcal{B}[\![b]\!]$ ein und betrachten die Mengenfunktion Γ mit:

$$\begin{aligned}
\Gamma(\varphi) = \ & \{(\sigma,\sigma'') \mid \exists \sigma'.\ (\sigma,\sigma') \in \mathcal{C}[\![c]\!] \wedge \beta(\sigma') = \mathit{false} \wedge (\sigma',\sigma'') \in \varphi\} \cup \\
& \{(\sigma,\sigma') \mid (\sigma,\sigma') \in \mathcal{C}[\![c]\!] \wedge \beta(\sigma') = \mathit{true}\}
\end{aligned}$$

Nun geben wir ein Regelsystem R an, so daß $\hat{R}(\varphi)$ gerade $\Gamma(\varphi)$ ist. Da die Existenz kleinster Fixpunkte für den Operator $\hat{R}$ bereits gezeigt wurde (Vorlesung), ist somit auch die Existenz eines kleinsten Fixpunktes für Γ garantiert. Diesen kleinsten Fixpunkt verwenden wir zur Definition von $\mathcal{C}[\![w]\!]$. Das folgende Regelsystem R hat die gewünschte Eigenschaft:

$$\begin{aligned}
R = \ & \{\{(\sigma',\sigma'')\}/(\sigma,\sigma'') \mid (\sigma,\sigma') \in \mathcal{C}[\![c]\!] \wedge \beta(\sigma') = \mathit{false}\} \cup \\
& \{\emptyset/(\sigma,\sigma') \mid (\sigma,\sigma') \in \mathcal{C}[\![c]\!] \wedge \beta(\sigma') = \mathit{true}\}
\end{aligned}$$

Der Beweis für $\hat{R}(\varphi) = \Gamma(\varphi)$ besteht aus einer längeren aber offensichtlichen Kette prädikatenlogischer Umformungen. Wir definieren nun $\mathcal{C}[\![w]\!]$ wie folgt:

$$\mathcal{C}[\![w]\!] = \mathit{fix}(\Gamma)$$

und erhalten damit

$$\mathcal{C}[\![w]\!] = \mathit{fix}(\hat{R}) = \bigcup_{n\in\omega} \hat{R}^n(\emptyset) = \bigcup_{n\in\omega} \Gamma^n(\emptyset)\ .$$

Aufgabe 3 (Lösungsvorschlag)

Sei

$$E(f) \equiv f(k) = \begin{cases} 1 & \text{falls } k = 0 \\ f(k+2) & \text{falls } k = 1 \\ f(k-2) & \text{sonst} \end{cases}$$

(a) Die Eigenschaft E gilt für die Funktionen f_1, f_2 und f_3:

$$\begin{array}{rll} & f_1(0) & = 1 \\ & f_1(1) & = 1 = f_1(3) \\ k > 1: & f_1(k) & = 1 = f_1(k-2) \end{array}$$

$$\begin{array}{rll} & f_2(0) & = 1 \\ & f_2(1) & = (1 \bmod 2) + 1 = (3 \bmod 2) + 1 = f_2(3) \\ k > 1: & f_2(k) & = (k \bmod 2) + 1 = (k - 2 \bmod 2) + 1 = f_2(k-2) \end{array}$$

$$\begin{array}{rll} & f_3(0) & = 1 \\ & f_3(1) & = undef. = f_3(3) \\ k > 1: & f_3(k) & = \begin{cases} 1 & \text{falls } k \text{ gerade} \\ undef. & \text{sonst} \end{cases} \\ & & = \begin{cases} 1 & \text{falls } k-2 \text{ gerade} \\ undef. & \text{sonst} \end{cases} = f_3(k-2) \end{array}$$

(b) Die 'kleiner'-Ordnung auf Funktionen ist durch die Mengeninklusion definiert:

$$f \sqsubseteq g \Leftrightarrow f \subseteq g$$

Da die Mengenschreibweise umständlich ist, formen wir um:

$$\begin{array}{ll} & f \subseteq g \\ \Leftrightarrow & \forall (x,y).\ (x,y) \in f \Rightarrow (x,y) \in g \\ \Leftrightarrow & \forall x\ y.\ f(x) = y \Rightarrow g(x) = y \\ \Leftrightarrow & \forall x\ y.\ f(x) = y \Rightarrow g(x) = f(x) \\ \Leftrightarrow & \forall x.\ (\neg \exists y. f(x) = y) \vee g(x) = f(x) \\ \Leftrightarrow & \forall x.\ f(x) = undef. \vee g(x) = f(x) \end{array}$$

Die kleinste Funktion, die E erfüllt, ist f_3. d.h.

$$\forall g.\ E(g) \Rightarrow f_3 \sqsubseteq g$$

Beweis:
Sei g beliebig und gelte $E(g)$.
Zu zeigen ist:

$$\begin{array}{ll} & f_3 \sqsubseteq g \\ \Leftrightarrow & \qquad \{\text{Definition}\} \\ & \forall x. f_3(x) = undef. \vee g(x) = f_3(x) \end{array}$$

Dies zeigen wir durch Fallunterscheidung:

i) x gerade, d.h. $\exists n.\ x = 2 \cdot n$:
Behauptung: $g(2 \cdot n) = f_3(2 \cdot n) = 1$
Beweis durch Induktion über n:
Induktionsanfang $n = 0$:

$$g(0) \stackrel{E(g)}{=} 1$$

Induktionsschritt $n \leadsto n+1$*:*
Gelte die *Induktionsvoraussetzung* (IV):

$$g(2 \cdot n) = 1$$

Dann gilt:

$$g(2 \cdot (n+1)) = g(2 \cdot n + 2) \overset{E(g)}{=} g(2 \cdot n) \overset{IV}{=} 1$$

ii) x ungerade:
Dann gilt $f_3(x) = undef$.