Korrekte Software: Grundlagen und Methoden

Vorlesung 3 vom 11.04.19

Denotationale Semantik

Serge Autexier, Christoph Lüth

Universität Bremen

Sommersemester 2019

Fahrplan

- Einführung
- Operationale Semantik
- Denotationale Semantik
- Äquivalenz der Operationalen und Denotationalen Semantik
- Der Floyd-Hoare-Kalkül
- Invarianten und die Korrektheit des Floyd-Hoare-Kalküls
- Strukturierte Datentypen
- Verifikationsbedingungen
- Vorwärts mit Floyd und Hoare
- Modellierung
- Spezifikation von Funktionen
- Referenzen und Speichermodelle
- Funktionsaufrufe und das Framing-Problem
- Ausblick und Rückblick

Überblick

- Kleinster Fixpunkt
- Denotationale Semantik für C0

Fixpunkt

- Sei $f : A \rightharpoonup A$ eine partielle Funktion. Ein **Fixpunkt** von f ist ein $a \in A$, so dass $f(a) = a$.
- Beispiele
  - Fixpunkte von $f(x) = \sqrt{x}$ sind 0 und 1; ebenfalls für $f(x) = x^2$.
  - Für die Sortierfunktion sind alle sortierten Listen Fixpunkte

Regeln und Regelinstanzen

Definition

Sei R eine Menge von Regeln $\frac{x_1 \ \ldots x_n}{y}$, $n \geq 0$.
Die Anwendung einer Regel auf spezifische $a_1 \ldots a_n$ ist eine Regelinstanz

- Betrachte folgende Regelmenge R

$$\frac{-}{2^2} \qquad \frac{-}{2^3} \qquad \frac{n \quad m}{n \cdot m}$$

- Regelinstanzen sind

$$\frac{-}{4} \qquad \frac{-}{8} \qquad \frac{4 \quad 8}{32} \qquad \frac{4 \quad 4}{16}$$

$$\frac{16 \quad 32}{512} \qquad \frac{3 \quad 5}{15} \qquad \ldots$$

Induktive Definierte Mengen

Definition

Seit R eine Menge von Regelinstanzen und B eine Menge. Dann definieren wir

$$\hat{R}(B) = \{y \mid \exists x_1, \ldots, x_k \subseteq B.\ \frac{x_1, \ldots, x_k}{y} \in R\} \text{ und}$$

$$\hat{R}^0(B) = B \text{ und } \hat{R}^{i+1}(B) = \hat{R}(\hat{R}^i(B))$$

Beispiel

- Betrachte folgende Regelmenge R

$$\frac{-}{2^2} \qquad \frac{-}{2^3} \qquad \frac{n \quad m}{n \cdot m}$$

- Was sind

$$\begin{aligned}
\hat{R}^0(\emptyset) &= \emptyset \\
\hat{R}^1(\emptyset) &= \hat{R}(\emptyset) = \{4, 8\} \\
\hat{R}^2(\emptyset) &=? \\
\hat{R}^3(\emptyset) &=? \\
\hat{R}^{i+1}(\emptyset) &=?
\end{aligned}$$

Beispiel

- Betrachte folgende Regelmenge R

$$\frac{-}{2^2} \qquad \frac{-}{2^3} \qquad \frac{n \quad m}{n \cdot m}$$

- Was sind

$$\begin{aligned}
\hat{R}^0(\emptyset) &= \emptyset \\
\hat{R}^1(\emptyset) &= \hat{R}(\emptyset) = \{4, 8\} \\
\hat{R}^2(\emptyset) &= \{16, 32, 64, 4, 8\} \\
\hat{R}^3(\emptyset) &= ? \\
\hat{R}^{i+1}(\emptyset) &= ?
\end{aligned}$$

Beispiel

- Betrachte folgende Regelmenge R

$$\frac{-}{2^2} \qquad \frac{-}{2^3} \qquad \frac{n \quad m}{n \cdot m}$$

- Was sind

$$\begin{aligned}
\hat{R}^0(\emptyset) &= \emptyset \\
\hat{R}^1(\emptyset) &= \hat{R}(\emptyset) = \{4, 8\} \\
\hat{R}^2(\emptyset) &= \{16, 32, 64, 4, 8\} \\
\hat{R}^3(\emptyset) &= \{128, 256, 512, 1024, 2048, 4096, 16, 32, 64, 4, 8\} \\
\hat{R}^{i+1}(\emptyset) &= ?
\end{aligned}$$

Beispiel

- Betrachte folgende Regelmenge R

$$\frac{-}{2^2} \qquad \frac{-}{2^3} \qquad \frac{n \quad m}{n \cdot m}$$

- Was sind

$$\begin{aligned}
\hat{R}^0(\emptyset) &= \emptyset \\
\hat{R}^1(\emptyset) &= \hat{R}(\emptyset) = \{4, 8\} \\
\hat{R}^2(\emptyset) &= \{16, 32, 64, 4, 8\} \\
\hat{R}^3(\emptyset) &= \{128, 256, 512, 1024, 2048, 4096, 16, 32, 64, 4, 8\} \\
\hat{R}^{i+1}(\emptyset) &= \{2^{2k+3l} | 1 \leq k + l \leq 2^i\}
\end{aligned}$$

Induktive Definierte Mengen

Definition

Seit R eine Menge von Regelinstanzen und B eine Menge. Dann definieren wir

$$\hat{R}(B) = \{y \mid \exists x_1, \ldots, x_k \subseteq B.\ \frac{x_1, \ldots, x_k}{y} \in R\} \text{ und}$$

$$\hat{R}^0(B) = B \text{ und } \hat{R}^{i+1}(B) = \hat{R}(\hat{R}^i(B))$$

Definition (Abgeschlossen und Monoton)

- Eine Menge S is **abgeschlossen unter R (R-abgeschlossen)** gdw. $\hat{R}(S) \subseteq S$
- Eine Operation f **ist monoton** gdw.

$$\forall A, B. A \subseteq B \Rightarrow f(A) \subseteq f(B)$$

Kleinster Fixpunkt Operator

Lemma

Für jede Menge von Regelinstanzen R ist die induzierte Operation $\hat{R}$ monoton.

Lemma

Sei $A_i = \hat{R}^i(\emptyset)$ für alle $i \in \mathbb{N}$ und $A = \bigcup_{i\in\mathbb{N}} A_i$. Dann gilt

a. *A ist R-abgeschlossen,*
b. *$\hat{R}(A) = A$, und*
c. *A ist die kleinste R-abgeschlossene Menge.*

Beweis von Lemma (a).

A ist R-abgeschlossen:

Sei $\frac{x_1,\ldots,x_k}{y} \in R$ und $x_1, \ldots, x_k \subseteq A$.
Da $A = \bigcup_{i\in\mathbb{N}} A_i$ gibt es ein j so dass $x_1, \ldots, x_k \subseteq A_j$.
Also auch:

$$\begin{aligned} y \in \hat{R}(A_j) &= \hat{R}(\hat{R}^j(\emptyset)) \\ &= \hat{R}^{j+1}(\emptyset) \\ &= A_{j+1} \subseteq A. \end{aligned}$$

□

Beweis von Lemma (b): $\hat{R}(A) = A$.

- $\hat{R}(A) \subseteq A$:
  Da A R-abgeschlossen gilt auch $\hat{R}(A) \subseteq A$.
- $A \subseteq \hat{R}(A)$:
  Sei $y \in A$. Dann $\exists n > 0.\ y \in A_n$ und $y \notin A_{n-1}$.
  Folglich muss es eine Regelinstanz $\frac{x_1,\ldots,x_k}{y} \in R$ geben mit
  $x_1, \ldots, x_k \subseteq A_{n-1} \subseteq A$.
  Da $\hat{R}$ monoton gilt $\hat{R}(A_{n-1}) \subseteq \hat{R}(A)$.
  Da $y \in A_n = \hat{R}(A_{n-1})$ folgt daraus $y \in \hat{R}(A)$.

$\square$

Beweis von Lemma (c).

A ist die kleinste R-abgeschlossene Menge, d.h. für jede R-abgeschlossene Menge B gilt $A \subseteq B$.

Beweis per Induktion über n dass gilt $A_n \subseteq B$:

- Basisfall:
  $A_0 = \emptyset \subseteq B$
- Induktionsschritt:
  Da B R-abgeschlossen ist gilt: $\hat{R}(B) \subseteq B$.
  Induktionsannahme: $A_n \subseteq B$.
  Dann gilt $A_{n+1} = \hat{R}(A_n) \subseteq \hat{R}(B) \subseteq B$ weil $\hat{R}$ monoton und B ist R-abgeschlossen.

$\square$

Kleinster Fixpunkt Operator

Definition

$$fix(\hat{R}) = \bigcup_{n \in N} \hat{R}^n(\emptyset)$$

ist der **kleinste Fixpunkt**.

Kleinster Fixpunkt

- Betrachte folgende Regelmenge R

$$\frac{-}{2^2} \qquad \frac{-}{2^3} \qquad \frac{n \quad m}{n \cdot m}$$

- Was sind

$$\begin{aligned}
\hat{R}^1(\emptyset) &= \hat{R}(\emptyset) = \{4, 8\} \\
\hat{R}^2(\emptyset) &= ? \\
\hat{R}^3(\emptyset) &= ? \\
\hat{R}^{i+1}(\emptyset) &= ?
\end{aligned}$$

Kleinster Fixpunkt

- Betrachte folgende Regelmenge R

$$\frac{-}{2^2} \qquad \frac{-}{2^3} \qquad \frac{n \quad m}{n \cdot m}$$

- Was sind

$$\begin{aligned} \hat{R}^1(\emptyset) &= \hat{R}(\emptyset) = \{4, 8\} \\ \hat{R}^2(\emptyset) &= ? \\ \hat{R}^3(\emptyset) &= ? \\ \hat{R}^{i+1}(\emptyset) &= ? \end{aligned}$$

- Wie sieht $fix(\hat{R})$ aus?

Denotationale Semantik — Motivation

- **Operationale Semantik**
  Eine Menge von Regeln, die einen Zustand und ein Programm in einen neuen Zustand oder Fehler überführen

$$\langle c, \sigma \rangle \to_{Stmt} \sigma' | \bot$$

- **Denotationale Semantik**
  Eine Menge von Regeln, die ein Programm in eine $\underbrace{\textbf{partielle Funktion}}_{\text{Denotat}}$ von Zustand nach Zustand überführen

$$\mathcal{C}[\![c]\!] : \Sigma \rightharpoonup \Sigma$$

Denotationale Semantik — Motivation

Zwei Programme sind äquivalent gdw. sie immer zum selben Zustand (oder Fehler) auswerten

$$c_0 \sim c_1 \text{ iff } (\forall \sigma, \sigma'. \langle c_0, \sigma \rangle \to_{Stmt} \sigma' \Leftrightarrow \langle c_1, \sigma \rangle \to_{Stmt} \sigma')$$

oder

Zwei Programme sind äquivalent gdw. sie dieselbe partielle Funktion **denotieren**

$$c_0 \sim c_1 \text{ iff } \{(\sigma, \sigma') | \langle c_0, \sigma \rangle \to_{Stmt} \sigma'\} = \{(\sigma, \sigma') | \langle c_1, \sigma \rangle \to_{Stmt} \sigma'\}$$

Denotierende Funktionen

- jeder a :**Aexp** denotiert eine partielle Funktion $\Sigma \rightharpoonup \mathbb{Z}$
- jeder b :**Bexp** denotiert eine partielle Funktion $\Sigma \rightharpoonup \mathbb{B}$
- jedes c :**Stmt** denotiert eine partielle Funktion $\Sigma \rightharpoonup \Sigma$

Definition (Partielle Funktion)

Eine **partielle Funktion** $f : X \rightharpoonup Y$ ist eine Relation $f \subseteq X \times Y$ so dass wenn $(x, y_1) \in f$ und $(x, y_2) \in f$ dann $y_1 = y_2$ (**Rechtseindeutigkeit**)

Notation: für $f : X \rightharpoonup Y$, $(x, y) \in f \iff f(x) = y$.

Denotat von Aexp

$$\mathcal{A}[\![a]\!] : \textbf{Aexp} \to (\Sigma \rightharpoonup \mathbb{Z})$$

$$\begin{array}{rcl}
\mathcal{A}[\![n]\!] & = & \{(\sigma, n) \mid \sigma \in \Sigma\} \\
\mathcal{A}[\![x]\!] & = & \{(\sigma, \sigma(x)) \mid \sigma \in \Sigma, x \in Dom(\sigma)\} \\
\mathcal{A}[\![a_0 + a_1]\!] & = & \{(\sigma, n_0 + n_1) \mid (\sigma, n_0) \in \mathcal{A}[\![a_0]\!] \wedge (\sigma, n_1) \in \mathcal{A}[\![a_1]\!]\} \\
\mathcal{A}[\![a_0 - a_1]\!] & = & \{(\sigma, n_0 - n_1) \mid (\sigma, n_0) \in \mathcal{A}[\![a_0]\!] \wedge (\sigma, n_1) \in \mathcal{A}[\![a_1]\!]\} \\
\mathcal{A}[\![a_0 * a_1]\!] & = & \{(\sigma, n_0 * n_1) \mid (\sigma, n_0) \in \mathcal{A}[\![a_0]\!] \wedge (\sigma, n_1) \in \mathcal{A}[\![a_1]\!]\} \\
\mathcal{A}[\![a_0 / a_1]\!] & = & \{(\sigma, n_0 / n_1) \mid (\sigma, n_0) \in \mathcal{A}[\![a_0]\!] \wedge (\sigma, n_1) \in \mathcal{A}[\![a_1]\!] \wedge n_1 \neq 0\}
\end{array}$$

Denotat von Bexp

$$\mathcal{B}[\![a]\!] : \textbf{Bexp} \to (\Sigma \rightharpoonup \mathbb{B})$$

$$\begin{aligned}
\mathcal{B}[\![1]\!] &= \{(\sigma, \mathit{true}) \mid \sigma \in \Sigma\} \\
\mathcal{B}[\![0]\!] &= \{(\sigma, \mathit{false}) \mid \sigma \in \Sigma\} \\
\mathcal{B}[\![a_0 == a_1]\!] &= \{(\sigma, \mathit{true}) \mid \sigma \in \Sigma, (\sigma, n_0) \in \mathcal{A}[\![a_0]\!](\sigma), \\
&\qquad (\sigma, n_1) \in \mathcal{A}[\![a_1]\!], n_0 = n_1\} \\
&\quad \cup \{(\sigma, \mathit{false}) \mid \sigma \in \Sigma, (\sigma, n_0) \in \mathcal{A}[\![a_0]\!](\sigma), \\
&\qquad (\sigma, n_1) \in \mathcal{A}[\![a_1]\!], n_0 \neq n_1\} \\
\mathcal{B}[\![a_0 < a_1]\!] &= \{(\sigma, \mathit{true}) \mid \sigma \in \Sigma, (\sigma, n_0) \in \mathcal{A}[\![a_0]\!](\sigma), \\
&\qquad (\sigma, n_1) \in \mathcal{A}[\![a_1]\!], n_0 < n_1\} \\
&\quad \cup \{(\sigma, \mathit{false}) \mid \sigma \in \Sigma, (\sigma, n_0) \in \mathcal{A}[\![a_0]\!](\sigma), \\
&\qquad (\sigma, n_1) \in \mathcal{A}[\![a_1]\!], n_0 \geq n_1\}
\end{aligned}$$

Denotat von Bexp

$$\mathcal{B}[\![a]\!] : \textbf{Bexp} \to (\Sigma \rightharpoonup \mathbb{B})$$

$$\begin{aligned}
\mathcal{B}[\![!b]\!] &= \{(\sigma, \mathit{true}) \mid \sigma \in \Sigma, (\sigma, \mathit{false}) \in \mathcal{B}[\![b]\!]\} \\
&\quad \cup \{(\sigma, \mathit{false}) \mid \sigma \in \Sigma, (\sigma, \mathit{true}) \in \mathcal{B}[\![b]\!]\} \\
\mathcal{B}[\![b_1 \ \&\& \ b2]\!] &= \{(\sigma, \mathit{false}) \mid \sigma \in \Sigma, (\sigma, \mathit{false}) \in \mathcal{B}[\![b_1]\!]\} \\
&\quad \cup \{(\sigma, t_2) \mid \sigma \in \Sigma, (\sigma, \mathit{true}) \in \mathcal{B}[\![b_1]\!], (\sigma, t_2) \in \mathcal{B}[\![b_2]\!]\} \\
\mathcal{B}[\![b_1 \ || \ b_2]\!] &= \{(\sigma, \mathit{true}) \mid \sigma \in \Sigma, (\sigma, \mathit{true}) \in \mathcal{B}[\![b_1]\!]\} \\
&\quad \cup \{(\sigma, t_2) \mid \sigma \in \Sigma, (\sigma, \mathit{false}) \in \mathcal{B}[\![b_1]\!], (\sigma, t_2) \in \mathcal{B}[\![b_2]\!]\}
\end{aligned}$$

Denotat von Stmt

$$\mathcal{C}[\![.]\!] : \textbf{Stmt} \to (\Sigma \rightharpoonup \Sigma)$$

$$
\begin{array}{rcl}
\mathcal{C}[\![x = a]\!] & = & \{(\sigma, \sigma[n/x]) \mid \sigma \in \Sigma \land (\sigma, n) \in \mathcal{A}[\![a]\!]\} \\
\mathcal{C}[\![c_1; c_2]\!] & = & \mathcal{C}[\![c_2]\!] \circ \mathcal{C}[\![c_1]\!] \qquad \text{Komposition von Relationen} \\
\mathcal{C}[\![\{\,\}]\!] & = & \textbf{Id} \qquad\qquad\qquad \textbf{Id} := \{(\sigma, \sigma) | \sigma \in \Sigma\} \\
\mathcal{C}[\![\textbf{if } (b)\ c_0\ \textbf{ else }\ c_1]\!] & = & \{(\sigma, \sigma') \mid (\sigma, \mathit{true}) \in \mathcal{B}[\![b]\!] \land (\sigma, \sigma') \in \mathcal{C}[\![c_0]\!]\} \\
 & & \cup \{(\sigma, \sigma') \mid (\sigma, \mathit{false}) \in \mathcal{B}[\![b]\!] \land (\sigma, \sigma') \in \mathcal{C}[\![c_1]\!]\}
\end{array}
$$

Denotat von Stmt

$$\mathcal{C}[\![.]\!] : \textbf{Stmt} \rightarrow (\Sigma \rightharpoonup \Sigma)$$

$$\begin{array}{rcll}
\mathcal{C}[\![x = a]\!] & = & \{(\sigma, \sigma[n/x]) \mid \sigma \in \Sigma \wedge (\sigma, n) \in \mathcal{A}[\![a]\!]\} & \\
\mathcal{C}[\![c_1 ; c_2]\!] & = & \mathcal{C}[\![c_2]\!] \circ \mathcal{C}[\![c_1]\!] & \text{Komposition von Relationen} \\
\mathcal{C}[\![\{\,\}]\!] & = & \textbf{Id} & \textbf{Id} := \{(\sigma, \sigma) | \sigma \in \Sigma\} \\
\mathcal{C}[\![\textbf{if}\ (b)\ c_0\ \textbf{else}\ c_1]\!] & = & \{(\sigma, \sigma') \mid (\sigma, \mathit{true}) \in \mathcal{B}[\![b]\!] \wedge (\sigma, \sigma') \in \mathcal{C}[\![c_0]\!]\} & \\
 & & \cup \{(\sigma, \sigma') \mid (\sigma, \mathit{false}) \in \mathcal{B}[\![b]\!] \wedge (\sigma, \sigma') \in \mathcal{C}[\![c_1]\!]\} &
\end{array}$$

Aber was ist

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

Denotationale Semantik für while

Sei $w \equiv \textbf{while}\ (b)\ c$ (und $\sigma \in \Sigma$). Operational gilt:

$$\begin{array}{rcl} w & \sim & \textbf{if}\ (b)\ \{c; w\}\ \textbf{else}\ \{\,\} \\ \mathcal{C}[\![w]\!] & \stackrel{?}{=} & \mathcal{C}[\![\textbf{if}\ (b)\ \{c; w\}\ \textbf{else}\ \{\,\}]\!] \end{array}$$

Denotationale Semantik für while

Sei $w \equiv$ **while** $(b)\ c$ (und $\sigma \in \Sigma$). Operational gilt:

$$\begin{aligned} w &\sim \textbf{if}\ (b)\ \{c; w\}\ \textbf{else}\ \{\,\} \\ \mathcal{C}[\![w]\!] &\stackrel{?}{=} \mathcal{C}[\![\textbf{if}\ (b)\ \{c; w\}\ \textbf{else}\ \{\,\}]\!] \end{aligned}$$

Konstruktion: Auffalten der Schleife

$$\Gamma(s) \stackrel{def}{=} \mathcal{C}[\![\textbf{if}\ (b)\ \{c; s\}\ \textbf{else}\ \{\,\}]\!]$$
$$\Gamma^0(s) \stackrel{def}{=} s, \Gamma^{i+1}(s) \stackrel{def}{=} \Gamma(\Gamma_i(s))$$

Denotationale Semantik für while

Sei $w \equiv \textbf{while}\ (b)\ c$ (und $\sigma \in \Sigma$). Operational gilt:

$$\begin{array}{rcl} w & \sim & \textbf{if}\ (b)\ \{c; w\}\ \textbf{else}\ \{\,\} \\ \mathcal{C}[\![w]\!] & \stackrel{?}{=} & \mathcal{C}[\![\textbf{if}\ (b)\ \{c; w\}\ \textbf{else}\ \{\,\}]\!] \end{array}$$

Konstruktion: Auffalten der Schleife

$$\Gamma(s) \stackrel{def}{=} \mathcal{C}[\![\textbf{if}\ (b)\ \{c; s\}\ \textbf{else}\ \{\,\}]\!]$$
$$\Gamma^0(s) \stackrel{def}{=} s, \Gamma^{i+1}(s) \stackrel{def}{=} \Gamma(\Gamma_i(s))$$

Semantik von w: Beliebig oft auffalten

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

Denotationale Semantik für while

Sei $w \equiv$ **while** $(b)\ c$ (und $\sigma \in \Sigma$). Operational gilt:

$$\begin{array}{rcl} w & \sim & \textbf{if}\ (b)\ \{c; w\}\ \textbf{else}\ \{\,\} \\ \mathcal{C}[\![w]\!] & \stackrel{?}{=} & \mathcal{C}[\![\textbf{if}\ (b)\ \{c; w\}\ \textbf{else}\ \{\,\}]\!] \end{array}$$

Konstruktion: Auffalten der Schleife

$$\Gamma(s) \stackrel{def}{=} \mathcal{C}[\![\textbf{if}\ (b)\ \{c; s\}\ \textbf{else}\ \{\,\}]\!]$$

$$\Gamma^0(s) \stackrel{def}{=} s, \Gamma^{i+1}(s) \stackrel{def}{=} \Gamma(\Gamma_i(s))$$

Semantik von w: Beliebig oft auffalten

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

Was ist **?**

Denotationale Semantik von while

Formale Konstruktion (s ist ein **Denotat**):

$$\Gamma(s) \stackrel{def}{=} \mathcal{C}[\![\textbf{if}\ (b)\ \{c; s\}\ \textbf{else}\ \{\ \}]\!]$$

$$\Gamma(s) \stackrel{def}{=} \{(\sigma, \sigma') \mid \exists \sigma''.\, (\sigma, \mathit{true}) \in \mathcal{B}[\![b]\!] \wedge (\sigma, \sigma'') \in \mathcal{C}[\![c]\!] \wedge (\sigma'', \sigma') \in s\}$$
$$\cup \{(\sigma, \sigma) \mid (\sigma, \mathit{false}) \in \mathcal{B}[\![b]\!]\}$$

Γ ist wie $\hat{R}$, mit R definiert wie folgt:

$$R = \left\{ \frac{(\sigma'', \sigma')}{(\sigma, \sigma')} \mid (\sigma, \mathit{true}) \in \mathcal{B}[\![b]\!] \wedge (\sigma, \sigma'') \in \mathcal{C}[\![c]\!] \right\}$$
$$\cup \left\{ \frac{}{(\sigma, \sigma)} \mid (\sigma, \mathit{false}) \in \mathcal{B}[\![b]\!] \right\}$$

Dann ist $\mathcal{C}[\![w]\!]$ der Fixpunkt von Γ:

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

Denotation für Stmt

$$\mathcal{C}[\![.]\!] : \textbf{Stmt} \to (\Sigma \rightharpoonup \Sigma)$$

$$\begin{array}{rcll}
\mathcal{C}[\![x = a]\!] & = & \{(\sigma, \sigma[n/x]) \mid \sigma \in \Sigma \land (\sigma, n) \in \mathcal{A}[\![a]\!]\} & \\
\mathcal{C}[\![c_1; c_2]\!] & = & \mathcal{C}[\![c_2]\!] \circ \mathcal{C}[\![c_1]\!] & \text{Komposition von Relationen} \\
\mathcal{C}[\![\{\,\}]\!] & = & \textbf{Id} & \textbf{Id} := \{(\sigma, \sigma) | \sigma \in \Sigma\} \\
\mathcal{C}[\![\textbf{if } (b)\ c_0\ \textbf{ else }\ c_1]\!] & = & \{(\sigma, \sigma') \mid (\sigma, \mathit{true}) \in \mathcal{B}[\![b]\!] \land (\sigma, \sigma') \in \mathcal{C}[\![c_0]\!]\} & \\
 & & \cup\, \{(\sigma, \sigma') \mid (\sigma, \mathit{false}) \in \mathcal{B}[\![b]\!] \land (\sigma, \sigma') \in \mathcal{C}[\![c_1]\!]\} & \\
\mathcal{C}[\![\textbf{while } (b)\ c]\!] & = & \mathit{fix}(\Gamma) &
\end{array}$$

mit

$$\begin{array}{rcl}
\Gamma(\psi) & = & \{(\sigma, \sigma') \mid (\sigma, \mathit{true}) \in \mathcal{B}[\![b]\!] \land (\sigma, \sigma') \in \psi \circ \mathcal{C}[\![c]\!]\} \\
 & & \cup\, \{(\sigma, \sigma) \mid (\sigma, \mathit{false}) \in \mathcal{B}[\![b]\!]\}
\end{array}$$

Der Fixpunkt bei der Arbeit

Beispielprogramme:

```
x= 0;
while (n > 0) {
   x= x+n;
   n= n-1;
   }
```

```
x= 0;
while (1) {
   x= x+1;
   }
```

```
x= 0;
while (n < 0) {
   x= x+1;
   }
```

Weitere Intuition zur Fixpunkt Konstruktion

- Sei $w \equiv \textbf{while}\ (b)\ c$
- Zur Erinnerung: Wir haben begonnen mit $w \sim \textbf{if}\ (b)\ \{c; w\}\ \textbf{else}\ \{\,\}$
- Dann müsste auch gelten

$$\mathcal{C}[\![w]\!] \stackrel{!}{=} \mathcal{C}[\![\textbf{if}\ (b)\ \{c; w\}\ \textbf{else}\ \{\,\}]\!]$$

  - Beweis an der Tafel.
- Es müsste ferner gelten

$$(\sigma, \sigma') \in \mathcal{C}[\![w]\!] \Longrightarrow (\sigma', \mathit{false}) \in \mathcal{B}[\![b]\!]$$

  - Beweis an der Tafel.

Zusammenfassung

- Die denotationale Semantik bildet Programme (Ausdrücke) auf **partielle Funktionen** $\Sigma \rightharpoonup \Sigma$ ab.
- Zentral ist der Begriff des **kleinsten Fixpunktes**, der die Semantik der while-Schleife bildet.
- Undefiniertheit wird **implizit** behandelt (durch die Partialität von $\Sigma \rightharpoonup \Sigma$).
  - Nicht-Termination und Undefiniertheit sind semantisch äquivalent.
- Genaues Verhältnis zur **operationalen Semantik?** Nächste Vorlesung