Semantikdefinitionsstile

Es gibt unterschiedliche Stile, die Semantik einer Programmiersprache festzulegen. Sie richten sich an unterschiedliche Adressaten und deren spezifische Sicht auf die Semantik...

Insbesondere unterscheiden wir den...

- *denotationellen*
- *operationellen*
- *axiomatischen*

Stil.

Semantikdefinitionsstile (2)

- *Sprachentwicklersicht*
  - Denotationelle Semantik
- *Sprach- und Anwendungsimplementierersicht*
  - Operationelle Semantik
    - Strukturell operationelle Semantik (small steps semantics)
    - Natürliche Semantik (big steps semantics)
- *Verifizierersicht*
  - Axiomatische Semantik

Strukturell operationelle Semantik

...am Beispiel von **WHILE**

$$[\text{skip}_{sos}] \quad \frac{}{\langle \mathbf{skip}, \sigma \rangle \Rightarrow \sigma}$$

$$[\text{abort}_{sos}] \quad \frac{}{\langle \mathbf{abort}, \sigma \rangle \Rightarrow \mathit{undef}}$$

$$[\text{ass}_{sos}] \quad \frac{}{\langle x := t, \sigma \rangle \Rightarrow \sigma[[\![t]\!]_A(\sigma)/x]}$$

$$[\text{comp}^1_{sos}] \quad \frac{\langle \Pi_1, \sigma \rangle \Rightarrow \langle \Pi_1', \sigma' \rangle}{\langle \Pi_1; \Pi_2, \sigma \rangle \Rightarrow \langle \Pi_1'; \Pi_2, \sigma' \rangle}$$

$$[\text{comp}^2_{sos}] \quad \frac{\langle \Pi_1, \sigma \rangle \Rightarrow \sigma'}{\langle \Pi_1; \Pi_2, \sigma \rangle \Rightarrow \langle \Pi_2, \sigma' \rangle}$$

Strukturell operationelle Semantik (2)

$$[\mathrm{if}^{tt}_{sos}] \quad \frac{}{\langle \textbf{if } b \textbf{ then } \Pi_1 \textbf{ else } \Pi_2 \textbf{ fi}, \sigma\rangle \Rightarrow \langle \Pi_1, \sigma\rangle} \qquad [\![b]\!]_B(\sigma) = \mathtt{tt}$$

$$[\mathrm{if}^{ff}_{sos}] \quad \frac{}{\langle \textbf{if } b \textbf{ then } \Pi_1 \textbf{ else } \Pi_2 \textbf{ fi}, \sigma\rangle \Rightarrow \langle \Pi_2, \sigma\rangle} \qquad [\![b]\!]_B(\sigma) = \mathtt{ff}$$

$$[\mathrm{while}_{sos}] \quad \frac{}{\langle \textbf{while } b \textbf{ do } \Pi \textbf{ od}, \sigma\rangle \Rightarrow \langle \textbf{if } b \textbf{ then } \Pi\textbf{; while } b \textbf{ do } \Pi \textbf{ od else skip fi}, \sigma\rangle}$$

Strukturell operationelle Semantik (3)

Der Fokus liegt auf...

- *individuellen Schritten* einer Berechnungsfolge, d.h. auf der Ausführung von Zuweisungen und Tests

  Intuitive Bedeutung der Transitionsrelation...

  $$\langle \Pi, \sigma \rangle \Rightarrow \gamma$$

  ...mit γ von der Form $\langle \Pi', \sigma' \rangle$ oder σ' oder $undef$ beschreibt den *ersten* Schritt der Berechnungsfolge von Π angesetzt auf σ. Folgende Übergänge sind möglich:

  - γ von der Form $\langle \Pi', \sigma' \rangle$:
    Abarbeitung von Π nicht vollständig; das Restprogramm Π' ist auf σ' anzusetzen

  - γ von der Form σ':
    Abarbeitung von Π vollständig; Π angesetzt auf σ terminiert in einem Schritt in σ'

  - γ von der Form $undef$:
    Abarbeitung von Π terminiert irregulär

Natürliche Semantik

...am Beispiel von **WHILE**

$$[\text{skip}_{ns}] \quad \frac{}{\langle \mathbf{skip}, \sigma \rangle \to \sigma}$$

$$[\text{abort}_{ns}] \quad \frac{}{\langle \mathbf{abort}, \sigma \rangle \to \mathit{undef}}$$

$$[\text{ass}_{ns}] \quad \frac{}{\langle x := t, \sigma \rangle \to \sigma[[\![t]\!]_A(\sigma)/x]}$$

$$[\text{comp}_{ns}] \quad \frac{\langle \Pi_1, \sigma \rangle \to \sigma', \langle \Pi_2, \sigma' \rangle \to \sigma''}{\langle \Pi_1 ; \Pi_2, \sigma \rangle \to \sigma''}$$

Natürliche Semantik (2)

$$[\mathrm{if}^{tt}_{ns}] \quad \frac{\langle \Pi_1, \sigma\rangle \to \sigma'}{\langle \textbf{if } b \textbf{ then } \Pi_1 \textbf{ else } \Pi_2 \textbf{ fi}, \sigma\rangle \to \sigma'} \qquad [\![b]\!]_B(\sigma) = \mathtt{tt}$$

$$[\mathrm{if}^{ff}_{ns}] \quad \frac{\langle \Pi_2, \sigma\rangle \to \sigma'}{\langle \textbf{if } b \textbf{ then } \Pi_1 \textbf{ else } \Pi_2 \textbf{ fi}, \sigma\rangle \to \sigma'} \qquad [\![b]\!]_B(\sigma) = \mathtt{ff}$$

$$[\mathrm{while}^{tt}_{ns}] \quad \frac{\langle \Pi, \sigma\rangle \to \sigma', \langle \textbf{while } b \textbf{ do } \Pi \textbf{ od}, \sigma'\rangle \to \sigma''}{\langle \textbf{while } b \textbf{ do } \Pi \textbf{ od}, \sigma\rangle \to \sigma''} \qquad [\![b]\!]_B(\sigma) = \mathtt{tt}$$

$$[\mathrm{while}^{ff}_{ns}] \quad \frac{}{\langle \textbf{while } b \textbf{ do } \Pi \textbf{ od}, \sigma\rangle \to \sigma} \qquad [\![b]\!]_B(\sigma) = \mathtt{ff}$$

Natürliche Semantik (3)

Der Fokus liegt auf...

- Zusammenhang von *initialem* und *finalem* Zustand einer Berechnungsfolge

  Intuitive Bedeutung von...

  $$\langle \Pi, \sigma \rangle \rightarrow \gamma$$

  ...mit γ von der Form σ' oder $undef$ ist: Π angesetzt auf initialen Zustand σ terminiert schließlich im finalen Zustand σ' bzw. terminiert irregulär.

Denotationelle Semantik

...am Beispiel von **WHILE**

$[\![skip]\!]_{ds} = id$

$[\![abort]\!]_{ds} = undef$

$[\![x := t]\!]_{ds}(\sigma) = \sigma[[\![t]\!]_A(\sigma)/x]$

$[\![\Pi_1;\ \Pi_2]\!]_{ds} = [\![\Pi_2]\!]_{ds} \circ [\![\Pi_1]\!]_{ds}$

$[\![\textbf{if } b \textbf{ then } \Pi_1 \textbf{ else } \Pi_2 \textbf{ fi}]\!]_{ds} = cond([\![b]\!]_{\mathcal{B}}, [\![\Pi_1]\!]_{ds}, [\![\Pi_2]\!]_{ds})$

$[\![\textbf{while } b \textbf{ do } \Pi \textbf{ od}]\!]_{ds} = FIX\ F$

where $F\ g = cond([\![b]\!]_{\mathcal{B}}, g \circ [\![\Pi]\!]_{ds}, id)$

Denotationelle Semantik (3)

- Operationelle Semantik
  ...der Fokus liegt darauf *wie* ein Programm ausgeführt wird

- Denotationelle Semantik
  ...der Fokus liegt auf dem *Effekt*, den die Ausführung eines Programms hat: Für jedes *syntaktische* Konstrukt gibt es eine *semantische* Funktion, die ersterem ein *mathematisches Objekt* zuweist, i.a. eine Funktion, die den Effekt der Ausführung des Konstrukts beschreibt.

  Zentral: **Kompositionalität**!

  Intuitiv:

  - Für jedes Element der elementaren syntaktischen Konstrukte/Kategorien gibt es eine zugehörige semantische Funktion

  - Für jedes Element eines zusammengesetzten syntaktischen Konstrukts/Kategorie gibt es eine semantische Funktion, die über die semantischen Funktionen der Komponenten des zusammengesetzten Konstrukts definiert ist.

Hauptergebnisse

Theorem

$$\forall\ \Pi\ \in\ \textbf{WHILE}.\ [\![\ \Pi\ ]\!]_{sos} = [\![\ \Pi\ ]\!]_{ns} = [\![\ \Pi\ ]\!]_{ds}$$