

## Denotationale Semantik

$S_{ds} : \text{Prog} \rightarrow (\text{State} \hookrightarrow \text{State})$

$\llbracket \text{assds} \rrbracket \quad S_{ds} \llbracket x := a \rrbracket s = s[x \mapsto \mathcal{A} \llbracket a \rrbracket s]$

$\llbracket \text{skipds} \rrbracket \quad S_{ds} \llbracket \text{skip} \rrbracket = \text{id}$

$\llbracket \text{compds} \rrbracket \quad S_{ds} \llbracket P_1; P_2 \rrbracket = S_{ds} \llbracket P_2 \rrbracket \circ S_{ds} \llbracket P_1 \rrbracket$

$\llbracket \text{compds} \rrbracket \quad S_{ds} \llbracket \text{if } b \text{ then } P_1 \text{ else } P_2 \rrbracket = \text{cond}(\mathcal{B} \llbracket b \rrbracket, S_{ds} \llbracket P_1 \rrbracket, S_{ds} \llbracket P_2 \rrbracket)$

$\llbracket \text{whileds} \rrbracket \quad S_{ds} \llbracket \text{while } b \text{ do } P \rrbracket = \text{FIX } F$

mit

$F : (\text{State} \hookrightarrow \text{State}) \rightarrow (\text{State} \hookrightarrow \text{State})$

$F g = \text{cond}(\mathcal{B} \llbracket b \rrbracket, g \circ S_{ds} \llbracket P \rrbracket, \text{id})$

43

## Bedeutung von FIX ? (1)

$\text{FIX} : ((\text{State} \hookrightarrow \text{State}) \rightarrow (\text{State} \hookrightarrow \text{State})) \rightarrow (\text{State} \hookrightarrow \text{State})$

$F : (\text{State} \hookrightarrow \text{State}) \rightarrow (\text{State} \hookrightarrow \text{State})$

$F g = \text{cond}(\mathcal{B} \llbracket b \rrbracket, g \circ S_{ds} \llbracket P \rrbracket, \text{id})$

$S_{ds} \llbracket \text{while } b \text{ do } P \rrbracket$

$= S_{ds} \llbracket \text{if } b \text{ then } (P; \text{while } b \text{ do } P) \text{ else skip} \rrbracket$

$= \text{cond}(\mathcal{B} \llbracket b \rrbracket, S_{ds} \llbracket P; \text{while } b \text{ do } P \rrbracket, S_{ds} \llbracket \text{skip} \rrbracket)$

$= \text{cond}(\mathcal{B} \llbracket b \rrbracket, S_{ds} \llbracket \text{while } b \text{ do } P \rrbracket \circ S_{ds} \llbracket P \rrbracket, \text{id})$

$= F(S_{ds} \llbracket \text{while } b \text{ do } P \rrbracket)$

daher:  $S_{ds} \llbracket \text{while } b \text{ do } P \rrbracket \text{ Fixpunkt von } F!$

45

## Notation

$\text{id } s = s$

$(f \circ g)s = \begin{cases} f(gs), & \text{falls } gs \neq \text{undef and } f(gs) \neq \text{undef} \\ \text{undef}, & \text{sonst} \end{cases}$

$\text{cond}(p, g_1, g_2)s = \begin{cases} g_1s, & \text{falls } ps = \text{t und } g_1s \neq \text{undef} \\ g_2s, & \text{falls } ps = \text{f und } g_2s \neq \text{undef} \\ \text{undef}, & \text{sonst} \end{cases}$

$\text{FIX } F = ???$

44

## Bedeutung von FIX ? (2)

$S_{ds} \llbracket \text{while } b \text{ do } P \rrbracket = \text{FIX } F$ , wobei  $F g = \text{cond}(\mathcal{B} \llbracket b \rrbracket, g \circ S_{ds} \llbracket P \rrbracket, \text{id})$

Zu klären:

- Besitzt  $F$  immer einen Fixpunkt?
- Kann  $F$  mehrere Fixpunkte haben?
- Wenn ja, welcher ist sinnvollerweise zu wählen?

Mögliche Ergebnisse der while-Schleife  $\text{while } b \text{ do } P$ :

- Sie terminiert.
- Lokale Nichtterminierung: Ein Teilkonstrukt im Schleifenumpf  $P$  terminiert nicht (in gewissen Zuständen).
- Globale Nichtterminierung: Die äußere while-Schleife terminiert nicht.

46

## Beispiele und intendierte Semantik (1)

Anfangszustand:  $s_0$  mit  $s_0 x = 5112002$

- $\text{while } 0 < x \text{ do } x := x - 1$
- $\text{while } 0 < x \text{ do if } x = 1 \text{ then (while true do skip) else } x := x - 1$
- $\text{while true do skip}$

47

## Beispiele und intendierte Semantik (b)

Fall (b): Schleifenrumpf von  $(\text{while } b \text{ do } P)$  terminiert nicht:

Es gibt Zustände  $s_1, \dots, s_n$  mit:

$$\bullet \mathcal{B}[b]_{s_i} = \begin{cases} \mathbf{t} & \text{falls } i < n \\ \mathbf{f} & \text{falls } i = n \end{cases}$$

$$\bullet \mathcal{S}_{ds} \llbracket P \rrbracket_{s_i} = \begin{cases} s_{i+1} & \text{für } i < n \\ \text{undef} & \text{für } i = n \end{cases}$$

Falls  $g$  Fixpunkt von  $F$  ist, so gilt

$$\text{für } i < n: \quad g s_i = g s_{i+1}$$

$$\text{für } i = n: \quad g s_n = (Fg) s_n$$

$$= \text{cond}(\mathcal{B}[b], g \circ \mathcal{S}_{ds} \llbracket P \rrbracket, \text{id}) s_n$$

$$= g(\mathcal{S}_{ds} \llbracket P \rrbracket s_n)$$

$$= \text{undef}$$

Also gilt für jeden Fixpunkt  $g_0$  von  $F$ :  $g_0 s_0 = \text{undef}$ .

49

## Beispiele und intendierte Semantik (a)

Fall (a):  $(\text{while } b \text{ do } P)$  terminiert in  $s_0$  (global):

Es gibt Zustände  $s_1, \dots, s_n$  mit:

$$\bullet \mathcal{B}[b]_{s_i} = \begin{cases} \mathbf{t} & \text{falls } i < n \\ \mathbf{f} & \text{falls } i = n \end{cases}$$

$$\bullet \mathcal{S}_{ds} \llbracket P \rrbracket_{s_i} = s_{i+1} \text{ für } i < n$$

Falls  $g$  Fixpunkt von  $F$  ist, so gilt

$$\text{für } i < n: \quad g s_i = (Fg) s_i$$

$$= \text{cond}(\mathcal{B}[b], g \circ \mathcal{S}_{ds} \llbracket P \rrbracket, \text{id}) s_i$$

$$= g(\mathcal{S}_{ds} \llbracket P \rrbracket s_i)$$

$$= g s_{i+1}$$

$$\text{für } i = n: \quad g s_n = (Fg) s_n$$

$$= \text{cond}(\mathcal{B}[b], g \circ \mathcal{S}_{ds} \llbracket P \rrbracket, \text{id}) s_n$$

$$= \text{id } s_n$$

$$= s_n$$

Also gilt für jeden Fixpunkt  $g_0$  von  $F$ :  $g_0 s_0 = s_n$ .

48

## Beispiele und intendierte Semantik (c)

Fall (c): Schleife  $(\text{while } b \text{ do } P)$  im Zustand  $s_0$  terminiert nicht:

Es gibt Zustände  $s_1, s_2, \dots$  mit:

$$\bullet \mathcal{B}[b]_{s_i} = \mathbf{t} \text{ für alle } i.$$

$$\bullet \mathcal{S}_{ds} \llbracket P \rrbracket_{s_i} = s_{i+1} \text{ für alle } i.$$

Falls  $g$  Fixpunkt von  $F$  ist, so gilt für alle  $i$ :

$$g s_i = g s_{i+1}$$

Also gilt für jeden Fixpunkt  $g_0$  von  $F$  lediglich:  $g_0 s_0 = g_0 s_i$  für alle  $i$ , aber der Wert von  $g_0 s_0$  bleibt unbestimmt!

→ Spielraum für die Wahl eines Fixpunktes

→ Intuition: Vermeide unnötige Festlegungen, wähle “kleinsten” Fixpunkt

50

## Intendierte Eigenschaften von FIX

$\mathcal{B}_{ds}[\llbracket \text{while } b \text{ do } P \rrbracket] = \text{FIX } F$ , wobei  $Fg = \text{cond}(\mathcal{B}[b], g \circ S_{ds}[\llbracket P \rrbracket], \text{id})$

Der intendierte Fixpunkt  $\text{FIX } F$  sollte eine partielle Funktion  $g_0 : \text{State} \hookrightarrow \text{State}$  sein mit:

- $g_0$  Fixpunkt von  $F$ :

$$F g_0 = g_0$$

- Falls  $g$  ebenfalls Fixpunkt von  $F$ , dann ist  $g$  *mindestens genauso definiert* wie  $g_0$ :

Falls  $F g = g$  und  $g_0 s = s'$ , dann gilt auch  $g s = s'$ ,  
für alle  $s, s'$ .

51

## Partialordnungen $(D, \sqsubseteq)$

$D, \sqsubseteq$  mit  $\sqsubseteq \subseteq D \times D$  *Partialordnung*<sup>a</sup>, falls

- *reflexiv*:  $d \sqsubseteq d$  für alle  $d \in D$
- *transitiv*: Falls  $d_1 \sqsubseteq d_2$  und  $d_2 \sqsubseteq d_3$ , dann auch  $d_1 \sqsubseteq d_3$ , für alle  $d_1, d_2, d_3 \in D$ .
- *anti-symmetrisch*: Falls  $d_1 \sqsubseteq d_2$  und  $d_2 \sqsubseteq d_1$ , dann folgt  $d_1 = d_2$ , für alle  $d_1, d_2 \in D$ .

**Lemma:** Falls  $(D, \sqsubseteq)$  ein *kleinstes* Element besitzt, dann ist dieses eindeutig (Bezeichnung:  $\perp$ ).

**Beispiel:** Sei  $S \neq \emptyset$  und  $\mathcal{P}(S) = \{T \mid T \subseteq S\}$ . Dann ist  $(\mathcal{P}(S), \subseteq)$  eine Partialordnung. Das kleinste Element existiert hier und ist  $\emptyset$ .

<sup>a</sup> englisch: partially ordered set = poset

53

## Eine Ordnung auf $\text{State} \hookrightarrow \text{State}$

*Definiertheitsordnung:*

$g_1 \sqsubseteq g_2$  gdw. für alle  $s, s'$  gilt: Falls  $g_1 s = s'$ , dann auch  $g_2 s = s'$ .

**Anforderungen an FIX formal:**

- $\text{FIX } F$  ist ein Fixpunkt von  $F$ :  
 $\text{FIX } F = F(\text{FIX } F)$
- $\text{FIX } F$  ist der kleinste Fixpunkt von  $F$ :  
Falls  $g = Fg$ , dann  $\text{FIX } F \sqsubseteq g$

52

## Ordnung für Semantik: $(\text{State} \hookrightarrow \text{State}, \sqsubseteq)$

**Definition:** (*Definiertheitsordnung*)

Die Partialordnung  $\sqsubseteq$  auf  $\text{State} \hookrightarrow \text{State}$  wird definiert durch

$g_1 \sqsubseteq g_2$  gdw.: Falls  $g_1 s = s'$ , dann auch  $g_2 s = s'$ , für alle  $s, s'$ .

**Satz:**

$(\text{State} \hookrightarrow \text{State}, \sqsubseteq)$  ist eine Partialordnung. Die partielle Funktion  $\perp$ , die definiert ist durch  $\perp s = \text{undef}$  für alle  $s$ , ist das kleinste Element von  $\text{State} \hookrightarrow \text{State}$ .

54

## Beispiel für Definiertheitsordnung

$g_1 s = s$  für alle  $s$

$$g_2 s = \begin{cases} s, & \text{falls } s \geq 0 \\ \text{undef}, & \text{sonst} \end{cases}$$

$$g_3 s = \begin{cases} s, & \text{falls } s \geq 0 \\ \text{undef}, & \text{sonst} \end{cases}$$

$$g_4 s = \begin{cases} s, & \text{falls } s \leq 0 \\ \text{undef}, & \text{sonst} \end{cases}$$

- $g_3 \sqsubseteq g_2 \sqsubseteq g_1, g_3 \sqsubseteq g_4 \sqsubseteq g_1$
- $g_2$  und  $g_4$  unvergleichbar

55

## Vollständige Verbände

$D, \sqsubseteq$  *vollständiger Verband* falls jede Teilmenge von  $D$  eine kleinste obere Schranke besitzt.

- **Übung:**  $\mathcal{P}(\{a, b, c\})$  ist ein vollständiger Verband.

Beweisidee: Kleinste obere Schranke ergibt sich als Vereinigung.

- **Übung:** (State  $\hookrightarrow$  State,  $\sqsubseteq$ ) ist **kein** vollständiger Verband.  
Grund: Nicht alle partiellen Funktionen  $g_1, g_2 : \text{State} \hookrightarrow \text{State}$  sind miteinander vergleichbar, d.h.:  $g_1 \not\sqsubseteq g_2$  und  $g_2 \not\sqsubseteq g_1$  ist möglich.

57

## Obere Schranken

Sei  $(D, \sqsubseteq)$  eine Partialordnung und sei  $Y \subseteq D, d \in D$ .

- $d$  *obere Schranke von*  $Y$ , falls

$$d \sqsubseteq d \text{ für alle } d \in Y$$

- $d$  *kleinste obere Schranke von*  $Y$ , falls

- $d$  obere Schranke von  $Y$ , und
- wenn  $d'$  obere Schranke von  $Y$ , dann  $d \sqsubseteq d'$

**Übung:** Falls  $Y$  eine kleinste obere Schranke besitzt, dann ist diese eindeutig und wird mit  $\bigsqcup Y$  bezeichnet.

56

## Kettenvollständige Partialordnungen

- $Y \subseteq D$  *Kette* in  $D$ , falls für alle  $d_1, d_2 \in Y$  gilt:  $d_1 \sqsubseteq d_2$  oder  $d_2 \sqsubseteq d_1$  (wechselseitige Vergleichbarkeit).

- $(D, \sqsubseteq)$  *kettenvollständige Partialordnung* (*ccpo*<sup>3</sup>), falls jede Kette in  $D$  eine kleinste obere Schranke besitzt.

**Satz:** (State  $\hookrightarrow$  State,  $\sqsubseteq$ ) ist eine kettenvollständige Partialordnung (ccpo). Die kleinste obere Schranke  $\bigsqcup Y$  einer Kette  $Y$  ist dabei gegeben durch

$$(\bigsqcup Y) s = \begin{cases} g s, & \text{falls } g s \neq \text{undef für ein } g \in Y \\ \text{undef}, & \text{sonst} \end{cases}$$

**Beweis(struktur):**

- $g_0 = \begin{cases} g s, & \text{falls } g s \neq \text{undef für ein } g \in Y \\ \text{undef}, & \text{sonst} \end{cases}$  ist wohldefiniert.

- $g_0$  ist obere Schranke von  $Y$ .
- $g_0$  ist die kleinste obere Schranke von  $Y$ .

<sup>3</sup>englisch: chain complete partially ordered set

58

# Monotone Funktionen

geg.:  $(D, \sqsubseteq), (D', \sqsubseteq')$  ccpo's und eine (totale) Funktion  $f: D \rightarrow D'$ .

heißt *monoton*, falls gilt:

Wenn  $d_1 \sqsubseteq d_2$ , dann auch  $f\ d_1 \sqsubseteq' f\ d_2$ , für alle  $d_1, d_2 \in D$ .

**Beispiel(e):**

$f_1, f_2: \mathcal{P}(\{a, b, c\}) \rightarrow \mathcal{P}(\{d, e\})$  mit

$X$	$f_1\ X$	$f_2\ X$
$\{a, b, c\}$	$\{d, e\}$	$\{d\}$
$\{a, b\}$	$\{d\}$	$\{d\}$
$\{a, c\}$	$\{d, e\}$	$\{d\}$
$\{b, c\}$	$\{d, e\}$	$\{e\}$
$\{a\}$	$\{d\}$	$\{d\}$
$\{b\}$	$\{d\}$	$\{e\}$
$\{c\}$	$\{e\}$	$\{e\}$
$\emptyset$	$\emptyset$	$\{e\}$

Hier:

$f_1$  **monoton**

$f_2$  **nicht** **monoton**

# Monotone Funktionen auf ccpo's

**Lemma:** Seien  $(D, \sqsubseteq), (D', \sqsubseteq')$  ccpo's und  $f: D \rightarrow D'$  eine monotone Funktion. Dann gilt:

- Falls  $Y$  eine Kette in  $D$  ist, so ist  $\{f\ d \mid d \in Y\}$  eine Kette in  $D'$ .

- Ferner gilt:  $\bigsqcup' \{f\ d \mid d \in Y\} \sqsubseteq' f(\bigsqcup Y)$ .

**Beweis(struktur):**

- Für  $Y = \emptyset$  folgt das Ergebnis aus  $\bot' \sqsubseteq' f\ \bot$ .
- Für  $Y \neq \emptyset$ :
  - $\{f\ d \mid d \in Y\}$  ist Kette in  $D'$
  - $\bigsqcup' \{f\ d \mid d \in Y\} \sqsubseteq' f(\bigsqcup Y)$