

Korrekte Software: Grundlagen und Methoden
Vorlesung 2 vom 09.04.19
Operationale 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

Zutaten

```
// GGT(A,B)
if (a == 0) r = b;
else {
    while (b != 0) {
        if (a <= b)
            b = b - a;
        else a = a - b;
    }
    r = a;
}
```

- ▶ Programme berechnen **Werte**
- ▶ Basierend auf
 - ▶ Werte sind **Variablen** zugewiesen
 - ▶ Evaluation von **Ausdrücken**
- ▶ Folgt dem Programmablauf

Unsere Programmiersprache

Wir betrachten einen Ausschnitt der Programmiersprache **C** (**C0**).

Ausbaustufe 1 kennt folgende Konstrukte:

- ▶ Typen: **int**;
- ▶ Ausdrücke: Variablen, Literale (für ganze Zahlen), arithmetische Operatoren (für ganze Zahlen), Relationen (`==`, `<`, `...`), boolsche Operatoren (`&&`, `||`);
- ▶ Anweisungen:
 - ▶ Fallunterscheidung (**if**...**else**...), Iteration (**while**), Zuweisung, Blöcke;
 - ▶ Sequenzierung und leere Anweisung sind implizit

C0: Ausdrücke und Anweisungen

Aexp $a ::= \mathbf{Z} \mid \mathbf{ldt} \mid a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \mid a_1 / a_2$

Bexp $b ::= \mathbf{1} \mid \mathbf{0} \mid a_1 == a_2 \mid a_1 < a_2 \mid !b \mid b_1 \&\& b_2 \mid b_1 || b_2$

Exp $e ::= a \mid b$

Stmt $c ::=$ **ldt** = **Exp**
| **if** (b) c_1 **else** c_2
| **while** (b) c
| $c_1; c_2$
| $\{ \}$

NB: Nicht die **konkrete** Syntax.

Eine Handvoll Beispiele

```
a = (3+y)*x+5*b;
```

```
a = ((3+y)*x)+(5*b);
```

```
a = 3+y*x+5*b;
```

```
p = 1;
```

```
c = 1;
```

```
while ( c <= n ) {
```

```
    p= p* c;
```

```
    c= c+ 1;
```

```
}
```

Semantik von C0

- ▶ Die (operationale) Semantik einer imperativen Sprache wie C0 ist ein **Zustandsübergang**: das System hat einen impliziten Zustand, der durch Zuweisung von **Werten** an **Adressen** geändert werden kann.

Systemzustände

- ▶ Ausdrücke werten zu **Werten** $\mathbf{V}$ (hier ganze Zahlen) aus.
- ▶ Adressen **Loc** sind hier Programmvariablen (Namen): **Loc** = **Idt**
- ▶ Ein **Systemzustand** bildet Adressen auf Werte ab: $\Sigma = \mathbf{Loc} \rightarrow \mathbf{V}$
- ▶ Ein Programm bildet einen Anfangszustand **möglicherweise** auf einen Endzustand ab (wenn es **terminiert**).

Partielle, endliche Abbildungen

Zustände sind **partielle, endliche Abbildungen** (finite partial maps)

$$f : X \rightarrow A$$

Notation:

- ▶ $f(x)$ für den Wert von x in f (*lookup*)
- ▶ $f(x) = \perp$ wenn x nicht in f (*undefined*)
- ▶ $f[n/x]$ für den Update an der Stelle x mit dem Wert n :

$$f[n/x](y) \stackrel{\text{def}}{=} \begin{cases} n & \text{if } x = y \\ f(y) & \text{otherwise} \end{cases}$$

- ▶ $\langle x \mapsto n, y \mapsto m \rangle$ u.ä. für konkrete Abbildungen.
- ▶ $\langle \rangle$ ist die leere (überall undefinierte Abbildung):

$$\langle \rangle(x) = \perp$$

Operationale Semantik: Arithmetische Ausdrücke

Ein arithmetischer Ausdruck a wertet unter gegebenen Zustand σ zu einer ganzen Zahl n (Wert) aus oder zu einem Fehler $\perp$.

► **Aexp** $a ::= \mathbf{Z} \mid \mathbf{Idt} \mid a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \mid a_1 / a_2$

$$\langle a, \sigma \rangle \rightarrow_{Aexp} n \mid \perp$$

Operationale Semantik: Arithmetische Ausdrücke

Ein arithmetischer Ausdruck a wertet unter gegebenen Zustand σ zu einer ganzen Zahl n (Wert) aus oder zu einem Fehler $\perp$.

► **Aexp** $a ::= \mathbf{Z} \mid \mathbf{Idt} \mid a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \mid a_1 / a_2$

$$\langle a, \sigma \rangle \rightarrow_{Aexp} n \mid \perp$$

Regeln:

$$\frac{}{\langle n, \sigma \rangle \rightarrow_{Aexp} n}$$

Operationale Semantik: Arithmetische Ausdrücke

Ein arithmetischer Ausdruck a wertet unter gegebenen Zustand σ zu einer ganzen Zahl n (Wert) aus oder zu einem Fehler $\perp$.

► **Aexp** $a ::= \mathbf{Z} \mid \mathbf{ldt} \mid a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \mid a_1 / a_2$

$$\langle a, \sigma \rangle \rightarrow_{Aexp} n \mid \perp$$

Regeln:

$$\frac{}{\langle n, \sigma \rangle \rightarrow_{Aexp} n}$$

$$\frac{x \in \mathbf{ldt}, x \in Dom(\sigma), \sigma(x) = v}{\langle x, \sigma \rangle \rightarrow_{Aexp} v}$$

$$\frac{x \in \mathbf{ldt}, x \notin Dom(\sigma)}{\langle x, \sigma \rangle \rightarrow_{Aexp} \perp}$$

Operationale Semantik: Arithmetische Ausdrücke

► **Aexp** $a ::= \mathbf{Z} \mid \mathbf{ldt} \mid a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \mid a_1 / a_2$
 $\langle a, \sigma \rangle \rightarrow_{Aexp} n \mid \perp$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad n_i \in \mathbb{Z}, n \text{ Summe } n_1 \text{ und } n_2}{\langle a_1 + a_2, \sigma \rangle \rightarrow_{Aexp} n}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad \text{falls } n_1 = \perp \text{ oder } n_2 = \perp}{\langle a_1 + a_2, \sigma \rangle \rightarrow_{Aexp} \perp}$$

Operationale Semantik: Arithmetische Ausdrücke

► **Aexp** $a ::= \mathbf{Z} \mid \mathbf{Idt} \mid a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \mid a_1 / a_2$
 $\langle a, \sigma \rangle \rightarrow_{Aexp} n \mid \perp$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad n_i \in \mathbb{Z}, n \text{ Summe } n_1 \text{ und } n_2}{\langle a_1 + a_2, \sigma \rangle \rightarrow_{Aexp} n}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad \text{falls } n_1 = \perp \text{ oder } n_2 = \perp}{\langle a_1 + a_2, \sigma \rangle \rightarrow_{Aexp} \perp}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad n_i \in \mathbb{Z}, n \text{ Diff. } n_1 \text{ und } n_2}{\langle a_1 - a_2, \sigma \rangle \rightarrow_{Aexp} n}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad \text{falls } n_1 = \perp \text{ oder } n_2 = \perp}{\langle a_1 - a_2, \sigma \rangle \rightarrow_{Aexp} \perp}$$

Operationale Semantik: Arithmetische Ausdrücke

► **Aexp** $a ::= \mathbf{Z} \mid \mathbf{Idt} \mid a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \mid a_1 / a_2$

$$\langle a, \sigma \rangle \rightarrow_{Aexp} n \mid \perp$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad n_i \in \mathbb{Z}, n \text{ Produkt } n_1 \text{ und } n_2}{\langle a_1 * a_2, \sigma \rangle \rightarrow_{Aexp} n}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad \text{falls } n_1 = \perp \text{ oder } n_2 = \perp}{\langle a_1 * a_2, \sigma \rangle \rightarrow_{Aexp} \perp}$$

Operationale Semantik: Arithmetische Ausdrücke

► **Aexp** $a ::= \mathbf{Z} \mid \mathbf{ldt} \mid a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \mid a_1 / a_2$

$$\langle a, \sigma \rangle \rightarrow_{Aexp} n \mid \perp$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad n_i \in \mathbb{Z}, n \text{ Produkt } n_1 \text{ und } n_2}{\langle a_1 * a_2, \sigma \rangle \rightarrow_{Aexp} n}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad \text{falls } n_1 = \perp \text{ oder } n_2 = \perp}{\langle a_1 * a_2, \sigma \rangle \rightarrow_{Aexp} \perp}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad n_i \in \mathbb{Z}, n_2 \neq 0, n \text{ Quotient } n_1, n_2}{\langle a_1 / a_2, \sigma \rangle \rightarrow_{Aexp} n}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad \text{falls } n_1 = \perp, n_2 = \perp \text{ oder } n_2 = 0}{\langle a_1 / a_2, \sigma \rangle \rightarrow_{Aexp} \perp}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\overline{\langle (x + y) * (x - y), \sigma \rangle} \rightarrow_{Aexp}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\overline{\langle x + y, \sigma \rangle \rightarrow_{Aexp}} \quad \overline{\langle x - y, \sigma \rangle \rightarrow_{Aexp}}}{\overline{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp}}}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6}{\langle x + y, \sigma \rangle \rightarrow_{Aexp}} \quad \frac{}{\langle x - y, \sigma \rangle \rightarrow_{Aexp}}}{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp}}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x + y, \sigma \rangle \rightarrow_{Aexp}} \quad \overline{\langle x - y, \sigma \rangle \rightarrow_{Aexp}}}{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp}}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x + y, \sigma \rangle \rightarrow_{Aexp} 11} \quad \frac{}{\langle x - y, \sigma \rangle \rightarrow_{Aexp}}}{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp}}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x + y, \sigma \rangle \rightarrow_{Aexp} 11} \quad \frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x - y, \sigma \rangle \rightarrow_{Aexp}}}{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp}}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x + y, \sigma \rangle \rightarrow_{Aexp} 11} \quad \frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x - y, \sigma \rangle \rightarrow_{Aexp} 1}}{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp}}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x + y, \sigma \rangle \rightarrow_{Aexp} 11} \quad \frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x - y, \sigma \rangle \rightarrow_{Aexp} 1}}{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp} 11}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x + y, \sigma \rangle \rightarrow_{Aexp} 11} \quad \frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x - y, \sigma \rangle \rightarrow_{Aexp} 1}}{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp} 11}$$

$$\frac{}{\langle (x * x) - (y * y), \sigma \rangle \rightarrow_{Aexp}}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x + y, \sigma \rangle \rightarrow_{Aexp} 11} \quad \frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x - y, \sigma \rangle \rightarrow_{Aexp} 1}}{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp} 11}$$

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle x, \sigma \rangle \rightarrow_{Aexp} 6}{\langle x * x, \sigma \rangle \rightarrow_{Aexp} 36}}{\langle (x * x) - (y * y), \sigma \rangle \rightarrow_{Aexp}}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x + y, \sigma \rangle \rightarrow_{Aexp} 11} \quad \frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x - y, \sigma \rangle \rightarrow_{Aexp} 1}}{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp} 11}$$

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle x, \sigma \rangle \rightarrow_{Aexp} 6}{\langle x * x, \sigma \rangle \rightarrow_{Aexp} 36} \quad \frac{\langle y, \sigma \rangle \rightarrow_{Aexp} 5 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle y * y, \sigma \rangle \rightarrow_{Aexp} 25}}{\langle (x * x) - (y * y), \sigma \rangle \rightarrow_{Aexp}}$$

Beispiel-Ableitungen

Sei $\sigma \stackrel{\text{def}}{=} \langle x \mapsto 6, y \mapsto 5 \rangle$.

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x + y, \sigma \rangle \rightarrow_{Aexp} 11} \quad \frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle x - y, \sigma \rangle \rightarrow_{Aexp} 1}}{\langle (x + y) * (x - y), \sigma \rangle \rightarrow_{Aexp} 11}$$

$$\frac{\frac{\langle x, \sigma \rangle \rightarrow_{Aexp} 6 \quad \langle x, \sigma \rangle \rightarrow_{Aexp} 6}{\langle x * x, \sigma \rangle \rightarrow_{Aexp} 36} \quad \frac{\langle y, \sigma \rangle \rightarrow_{Aexp} 5 \quad \langle y, \sigma \rangle \rightarrow_{Aexp} 5}{\langle y * y, \sigma \rangle \rightarrow_{Aexp} 25}}{\langle (x * x) - (y * y), \sigma \rangle \rightarrow_{Aexp} 11}$$

Operationale Semantik: Boolesche Ausdrücke

- **Bexp** $b ::= 0 \mid 1 \mid a_1 == a_2 \mid a_1 < a_2 \mid !b \mid b_1 \&\& b_2 \mid b_1 \parallel b_2$
- $$\langle b, \sigma \rangle \rightarrow_{Bexp} true \mid false \mid \perp$$

Regeln:

$$\overline{\langle 1, \sigma \rangle \rightarrow_{Bexp} true}$$

$$\overline{\langle 0, \sigma \rangle \rightarrow_{Bexp} false}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad n_i \neq \perp, n_1 \text{ und } n_2 \text{ gleich}}{\langle a_1 == a_2, \sigma \rangle \rightarrow_{Bexp} true}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad n_i \neq \perp, n_1 \text{ und } n_2 \text{ ungleich}}{\langle a_1 == a_2, \sigma \rangle \rightarrow_{Bexp} false}$$

$$\frac{\langle a_1, \sigma \rangle \rightarrow_{Aexp} n_1 \quad \langle a_2, \sigma \rangle \rightarrow_{Aexp} n_2 \quad n_1 = \perp \text{ or } n_2 = \perp}{\langle a_1 == a_2, \sigma \rangle \rightarrow_{Bexp} \perp}$$

Operationale Semantik: Boolesche Ausdrücke

- **Bexp** $b ::= 0 \mid 1 \mid a_1 == a_2 \mid a_1 < a_2 \mid !b \mid b_1 \&\& b_2 \mid b_1 \parallel b_2$
- $$\langle b, \sigma \rangle \rightarrow_{Bexp} true \mid false \mid \perp$$

Regeln:

$$\frac{\langle b, \sigma \rangle \rightarrow_{Bexp} true}{\langle !b, \sigma \rangle \rightarrow_{Bexp} false} \quad \frac{\langle b, \sigma \rangle \rightarrow_{Bexp} false}{\langle !b, \sigma \rangle \rightarrow_{Bexp} true} \quad \frac{\langle b, \sigma \rangle \rightarrow_{Bexp} \perp}{\langle !b, \sigma \rangle \rightarrow_{Bexp} \perp}$$

$$\frac{\langle b_1, \sigma \rangle \rightarrow_{Bexp} t_1 \quad \langle b_2, \sigma \rangle \rightarrow_{Bexp} t_2}{\langle b_1 \&\& b_2, \sigma \rangle \rightarrow_{Bexp} t}$$

wobei $t = true$ wenn $t_1 = t_2 = true$;
 $t = false$ wenn $t_1 = false$ oder $(t_1 = true$ und $t_2 = false)$;
 $t = \perp$ sonst

Operationale Semantik: Boolesche Ausdrücke

- **Bexp** $b ::= 0 \mid 1 \mid a_1 == a_2 \mid a_1 < a_2 \mid !b \mid b_1 \&\& b_2 \mid b_1 \parallel b_2$
- $$\langle b, \sigma \rangle \rightarrow_{Bexp} true \mid false \mid \perp$$

Regeln:

$$\frac{\langle b_1, \sigma \rangle \rightarrow_{Bexp} t_1 \quad \langle b_2, \sigma \rangle \rightarrow_{Bexp} t_2}{\langle b_1 \parallel b_2, \sigma \rangle \rightarrow_{Bexp} t}$$

wobei $t = false$ wenn $t_1 = t_2 = false$;
 $t = true$ wenn $t_1 = true$ oder ($t_1 = false$ und $t_2 = true$);
 $t = \perp$ sonst

Operationale Semantik: Anweisungen

► **Stmt** $c ::= \text{Idt} = \text{Exp} \mid \text{if } (b) \ c_1 \ \text{else} \ c_2 \mid \text{while } (b) \ c \mid c_1; c_2 \mid \{ \}$

Beispiel:

$$\langle c, \sigma \rangle \rightarrow_{\text{Stmt}} \sigma' \mid \perp$$

$$\langle x = 5, \sigma \rangle \rightarrow_{\text{Stmt}} \sigma'$$

wobei $\sigma'(x) = 5$ und $\sigma'(y) = \sigma(y)$ für alle $y \neq x$

Operationale Semantik: Anweisungen

► **Stmt** $c ::= \text{Idt} = \text{Exp} \mid \text{if } (b) \ c_1 \ \text{else} \ c_2 \mid \text{while } (b) \ c \mid c_1; c_2 \mid \{ \}$

Regeln:

$$\langle \{ \}, \sigma \rangle \rightarrow_{\text{Stmt}} \sigma$$

$$\frac{\langle a, \sigma \rangle \rightarrow_{\text{Aexp}} n \in \mathbb{Z}}{\langle x = a, \sigma \rangle \rightarrow_{\text{Stmt}} \sigma[n/x]}$$

$$\frac{\langle a, \sigma \rangle \rightarrow_{\text{Aexp}} \perp}{\langle x = a, \sigma \rangle \rightarrow_{\text{Stmt}} \perp}$$

$$\frac{\langle c_1, \sigma \rangle \rightarrow_{\text{Stmt}} \sigma' \neq \perp \quad \langle c_2, \sigma' \rangle \rightarrow_{\text{Stmt}} \sigma'' \neq \perp}{\langle c_1; c_2, \sigma \rangle \rightarrow_{\text{Stmt}} \sigma''}$$

$$\frac{\langle c_1, \sigma \rangle \rightarrow_{\text{Stmt}} \perp}{\langle c_1; c_2, \sigma \rangle \rightarrow_{\text{Stmt}} \perp}$$

$$\frac{\langle c_1, \sigma \rangle \rightarrow_{\text{Stmt}} \sigma' \neq \perp \quad \langle c_2, \sigma' \rangle \rightarrow_{\text{Stmt}} \perp}{\langle c_1; c_2, \sigma \rangle \rightarrow_{\text{Stmt}} \perp}$$

Operationale Semantik: Anweisungen

► Stmt $c ::= \text{Idt} = \text{Exp} \mid \text{if } (b) \text{ } c_1 \text{ else } c_2 \mid \text{while } (b) \text{ } c \mid c_1; c_2 \mid \{ \}$

Regeln:

$$\frac{\langle b, \sigma \rangle \rightarrow_{Bexp} \text{true} \quad \langle c_1, \sigma \rangle \rightarrow_{Stmt} \sigma'}{\langle \text{if } (b) \text{ } c_1 \text{ else } c_2, \sigma \rangle \rightarrow_{Stmt} \sigma'}$$

$$\frac{\langle b, \sigma \rangle \rightarrow_{Bexp} \text{false} \quad \langle c_2, \sigma \rangle \rightarrow_{Stmt} \sigma'}{\langle \text{if } (b) \text{ } c_1 \text{ else } c_2, \sigma \rangle \rightarrow_{Stmt} \sigma'}$$

$$\frac{\langle b, \sigma \rangle \rightarrow_{Bexp} \perp}{\langle \text{if } (b) \text{ } c_1 \text{ else } c_2, \sigma \rangle \rightarrow_{Stmt} \perp}$$

Operationale Semantik: Anweisungen

► **Stmt** $c ::= \text{Idt} = \text{Exp} \mid \text{if } (b) \ c_1 \ \text{else} \ c_2 \mid \text{while } (b) \ c \mid c_1; c_2 \mid \{ \}$

Regeln:

$$\frac{\langle b, \sigma \rangle \rightarrow_{Bexp} false}{\langle \text{while } (b) \ c, \sigma \rangle \rightarrow_{Stmt} \sigma}$$

$$\frac{\langle b, \sigma \rangle \rightarrow_{Bexp} true \quad \langle c, \sigma \rangle \rightarrow_{Stmt} \sigma' \quad \langle \text{while } (b) \ c, \sigma' \rangle \rightarrow_{Stmt} \sigma''}{\langle \text{while } (b) \ c, \sigma \rangle \rightarrow_{Stmt} \sigma''}$$

$$\frac{\langle b, \sigma \rangle \rightarrow_{Bexp} true \quad \langle c, \sigma \rangle \rightarrow_{Stmt} \perp}{\langle \text{while } (b) \ c, \sigma \rangle \rightarrow_{Stmt} \perp}$$

$$\frac{\langle b, \sigma \rangle \rightarrow_{Bexp} \perp}{\langle \text{while } (b) \ c, \sigma \rangle \rightarrow_{Stmt} \perp}$$

Beispiel

```
x = 1;  
while (y != 0) {  
  y = y - 1;  
  x = 2 * x;  
}  
// x = 2y  
 $\sigma \stackrel{\text{def}}{=} \langle y \mapsto 3 \rangle$ 
```

Äquivalenz arithmetischer Ausdrücke

Gegeben zwei Aexp a_1 und a_2

► Sind sie gleich?

$$a_1 \sim_{Aexp} a_2 \text{ gdw } \forall \sigma, n. \langle a_1, \sigma \rangle \rightarrow_{Aexp} n \Leftrightarrow \langle a_2, \sigma \rangle \rightarrow_{Aexp} n$$

$$(x * x) + 2 * x * y + (y * y) \quad \text{und} \quad (x + y) * (x + y)$$

► Wann sind sie gleich?

$$\forall \sigma, n. \langle a_1, \sigma \rangle \rightarrow_{Aexp} n \Leftrightarrow \langle a_2, \sigma \rangle \rightarrow_{Aexp} n$$

$$\begin{array}{lll} x * x & \text{und} & 8 * x + 9 \\ x * x & \text{und} & x * x + 1 \end{array}$$

Äquivalenz Boolescher Ausdrücke

Gegeben zwei Bexp-Ausdrücke b_1 and b_2

► Sind sie gleich?

$$b_1 \sim_{Bexp} b_2 \text{ iff } \forall \sigma, b. \langle b_1, \sigma \rangle \rightarrow_{Bexp} b \Leftrightarrow \langle b_2, \sigma \rangle \rightarrow_{Bexp} b$$

$A \ || \ (A \ \&\& \ B)$ und A

Beweisen

Zwei Programme c_0, c_1 sind äquivalent gdw. sie die gleichen Zustandsveränderungen bewirken. Formal definieren wir

Definition

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

Ein einfaches Beispiel:

Lemma

Sei $w \equiv \text{while } (b) \ c$ mit $b \in \mathbf{Bexp}$, $c \in \mathbf{Stmt}$.

Dann gilt: $w \sim \text{if } (b) \ \{c; w\} \text{ else } \{\}$

Beweis

Gegeben beliebiger Programmzustand σ . Zu zeigen ist, dass sowohl w also auch **if** (b) $\{c; w\}$ **else** $\{\}$ zu dem selben Programmzustand auswerten oder beide zu einem Fehler. Der Beweis geht per Fallunterscheidung über die Auswertung von Teilausdrücken bzw. Teilprogrammen.

$$\textcircled{1} \langle b, \sigma \rangle \rightarrow_{Bexp} \perp:$$

$$\begin{aligned} & \langle \textbf{while} (b) c, \sigma \rangle \rightarrow_{Stmt} \perp \\ & \langle \textbf{if} (b) \{c; w\} \textbf{else} \{\}, \sigma \rangle \rightarrow_{Stmt} \perp \end{aligned}$$

$$\textcircled{2} \langle b, \sigma \rangle \rightarrow_{Bexp} \textit{false}:$$

$$\begin{aligned} & \langle \textbf{while} (b) c, \sigma \rangle \rightarrow_{Stmt} \sigma \\ & \langle \textbf{if} (b) \{c; w\} \textbf{else} \{\}, \sigma \rangle \rightarrow_{Stmt} \langle \{\}, \sigma \rangle \rightarrow_{Stmt} \sigma \end{aligned}$$

Beweis II

③ $\langle b, \sigma \rangle \rightarrow_{Bexp} true$:

① $\langle c, \sigma \rangle \rightarrow_{Stmt} \sigma'$

$$\begin{aligned} & \overbrace{\langle \text{while } (b) \ c, \sigma \rangle}^w \rightarrow_{Stmt} \langle c, \sigma \rangle \rightarrow_{Stmt} \sigma' \\ & \qquad \qquad \qquad \langle w, \sigma' \rangle \rightarrow_{Stmt} \sigma'' \\ \langle \text{if } (b) \ \{c; w\} \ \text{else} \ \{\}, \sigma \rangle & \rightarrow_{Stmt} \langle \{c; w\}, \sigma \rangle \rightarrow_{Stmt} \langle c, \sigma \rangle \rightarrow_{Stmt} \sigma' \\ & \qquad \qquad \qquad \langle w, \sigma' \rangle \rightarrow_{Stmt} \sigma'' \end{aligned}$$

② $\langle c, \sigma \rangle \rightarrow_{Stmt} \perp$

$$\begin{aligned} & \overbrace{\langle \text{while } (b) \ c, \sigma \rangle}^w \rightarrow_{Stmt} \langle c, \sigma \rangle \rightarrow_{Stmt} \perp \\ \langle \text{if } (b) \ \{c; w\} \ \text{else} \ \{\}, \sigma \rangle & \rightarrow_{Stmt} \langle \{c; w\}, \sigma \rangle \rightarrow_{Stmt} \langle c, \sigma \rangle \rightarrow_{Stmt} \perp \end{aligned}$$

Zusammenfassung

- ▶ Operationale Semantik als ein Mittel zur Beschreibung der Semantik
- ▶ Auswertungsregeln arbeiten entlang der syntaktischen Struktur
- ▶ Werten Ausdrücke zu Werten aus und Programme zu Zuständen (zu gegebenen Zustand)
- ▶ Fragen zu Programmen: Gleichheit