

2. Übungsblatt

Ausgabe: 20.04.17

Abgabe: 27.04.17

2.1 Operationale Semantik von Ausdrücken

5 Punkte

In der Vorlesung wurde die operationale Semantik von C0 eingeführt. Geben Sie die Ableitung der Auswertung folgender Ausdrücke in der operationalen Semantik unter einem gegebenen Zustand σ an:

- $5 * (3 + x)$ mit $\sigma(x) = 7$
- $a \parallel (a \ \&\& \ b)$ mit $\sigma(a) = 1$ und b undefiniert
- $(a \ \&\& \ b) \parallel a$ mit $\sigma(a) = 1$ und b undefiniert

2.2 Programmgleichheit

10 Punkte

Beweisen Sie die Äquivalenz der folgenden beiden Programme w_1 und w_2 .

```
w1 ≡ x = 1;
      m = 2;
      while (m > 0) {
        x = x * 2;
        m = m - 1;
      }
      r = x;
      x = 1;

w2 ≡ x = 1;
      r = 1;
      m = 1;
      while (m <= 2) {
        r = r * 2;
        m = m + 1;
      }
      m = 0;
```

Gehen Sie dabei vor wie in der Vorlesung und zeigen Sie, dass für beliebige Anfangszustände σ jeweils ein zu $\sigma[1/x][0/m][4/r]$ äquivalenter Endzustand erreicht wird. Berücksichtigen Sie, dass folgende Äquivalenzen auf Zuständen gelten:

$$\forall \sigma, n, m, \forall X, Y. X \neq Y \Rightarrow \sigma[n/X][m/Y] = \sigma[m/Y][n/X]$$
$$\forall \sigma, n, m, \forall X. \sigma[n/X][m/X] = \sigma[m/X]$$

2.3 Bedingungen

5 Punkte

Wir betrachten die Syntax für C0 aus der Vorlesung:

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

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

Exp $e ::= \mathbf{Aexp} \mid \mathbf{Bexp}$

Stmt $c ::= \mathbf{Loc} = \mathbf{Exp}; \mid \mathbf{if} (b) c_1 \mathbf{else} c_2 \mid \mathbf{while} (b) c \mid \{c^*\}$

In C0 gibt es auch den ternären Bedingungsoperator $B ? E_1 : E_2$, sowohl für arithmetische als auch boolesche Ausdrücke.

- Erweitern Sie die Syntax für arithmetische Ausdrücke um den ternären Bedingungsoperator.
- Definieren Sie entsprechend die operationalen Semantik für den Bedingungsoperator durch Angabe der geeigneten Regeln. Achten Sie auf Vollständigkeit und Striktheit — wann ist der gesamte Ausdruck undefiniert?
- Erweitern Sie die Syntax auch für boolesche Ausdrücke um den ternären Bedingungsoperator und definieren Sie die operationale Semantik entsprechend.