
Softwaretechnik

<http://swt.informatik.uni-freiburg.de/node/94>
<http://proglang.informatik.uni-freiburg.de/teaching/swt/2008/>

Übungsblatt 3

2008-05-23

Aufgabe 1 (Verschmelzen von Linksets; 2 Punkte)

Seien folgende zwei Linksets gegeben:

$$L_1 \equiv x : \text{int} \mid (b \approx y : \text{int} \vdash x > y : \text{bool}), (y \approx \emptyset \vdash 5 : \text{int})$$
$$L_2 \equiv b : \text{bool}, z : \text{int} \mid (x \approx \emptyset \vdash \text{if } b \text{ then } z \text{ else } 0 : \text{int})$$

Verschmelzen Sie L_1 und L_2 ; d.h. berechnen Sie $L_1 + L_2$.

Aufgabe 2 (Linking; (3+3) Punkte)

(a) Linken Sie nachfolgendes Linkset L ; d.h. führen Sie Linkschritte \rightsquigarrow so lange wie möglich aus.

$$L \equiv z : \text{int} \mid (b \approx y : \text{bool}, x : \text{int} \vdash \text{if } y \text{ then } x \text{ else } z : \text{int})$$
$$(y \approx x : \text{int} \vdash x > 5 : \text{bool})$$
$$(x \approx \emptyset \vdash 6 : \text{int})$$

(b) Zeigen Sie, dass die Linkschritt-Relation \rightsquigarrow intramodulare Konsistenz nicht erhält. Finden Sie also ein Linkset L mit $\text{intra-checked}(L)$, $L \rightsquigarrow L'$, aber nicht $\text{intra-checked}(L')$.

Aufgabe 3 (Interfaces für Featherweight Java; 12 Punkte)

Erweitern Sie Featherweight Java um Interfaces. Als Anhaltspunkt sei hier die Syntax der erweiterten Sprache gegeben:

$$CL ::= \text{class } C \text{ extends } D \text{ implements } E_1, \dots \{ C_1 f_1; \dots K M_1 \dots \}$$
$$\quad \mid \text{interface } C \text{ extends } D_1, \dots \{ S_1; \dots \}$$
$$S ::= C m(C_1 x_1, \dots)$$

(K , M , t und v sind wie in der Vorlesung definiert.)

Die Metavariablen C , D und E stehen für Klassen- und Interfacenamen. Eine Klassendeklaration **class** C **extends** D **implements** $E_1, \dots \{ C_1 f_1; \dots K M_1 \dots \}$ gibt jetzt nicht mehr nur die Superklasse D an, sondern spezifiziert auch die Interfaces E_1, \dots , die C implementiert. Falls E_1, \dots leer ist, dann implementiert C kein Interface.

Eine Interfacedeklaration **interface** C **extends** $D_1, \dots \{ S_1; \dots \}$ führt ein neues Interface C ein. Dabei sind die D_1, \dots (möglicherweise leer) die Superinterfaces von C .

Die Metavariablen S stehen für Methodensignaturen. Eine solche Methodensignatur gibt dabei nur den Rückgabebetyp und die Argumenttypen einer Methode an; es wird kein Methodenrumpf definiert.

Erweitern Sie jetzt die Typregeln und möglicherweise auch die operationelle Semantik von Featherweight Java. Benutzen Sie dabei Ihr Wissen über Interfaces in Java. Ihre Erweiterung soll so klein wie möglich sein und auf den Regeln der Vorlesung aufbauen. (In den Regeln aus der Vorlesung spezifizieren Klassendeklarationen nicht die Liste der implementierten Interfaces. Sie können solche Regeln trotzdem in Ihrer Erweiterung verwenden, indem Sie annehmen, dass diese Liste der Interfaces E_1, \dots ist, wobei E_1, \dots Namen sind, die in der Regel sonst nicht vorkommen.)

Abgabe: 2008-05-30, 12 Uhr **vor** der Saalübung im HS 00-036, Geb. 101.