

Übung zu “Automatische Analyse und Verifikation von Programmen”

Dieses Übungsblatt wird am Montag, den 24. Juli, von 15:45–17:15 Uhr im Raum 0.124 besprochen.

Aufgabe 20 *Operationelle Semantik des π -Kalküls*

Bestimmen Sie das Transitionssystem, d.h. alle möglichen Übergänge, folgender Prozesse:

- (a) $P_1 = (\nu x)(\bar{x}(y).\mathbf{0} \mid x(z).\bar{z}\langle \rangle.\mathbf{0} \mid y().\mathbf{0}) \mid x(w).\mathbf{0}$
- (b) $P_2 = (\nu x)(!\bar{x}\langle y, u \rangle.\mathbf{0} \mid !x(z, w).\bar{z}\langle w \rangle.\mathbf{0}) \mid !y(v).\mathbf{0}$

Aufgabe 21 *Prozesse*

Definieren Sie Prozesse mit folgendem Verhalten:

- (a) Einen Prozess F (Forwarder), dessen einziger freier Name f ist. Über f empfängt dieser Prozess zwei Kanalnamen p_1 und p_2 . Der Name, der im nächsten Schritt über p_1 geschickt wird, soll von F empfangen und nach p_2 weitergeleitet werden.
- (b) Ein Prozess F' , der sich im wesentlichen wie F verhält, aber so, dass der “Weiterleitungsdienst” nach dem Empfang von p_1 und p_2 beliebig lange zur Verfügung steht.
- (c) Ein Server S , der auf zwei Kanälen e und z lauscht und auf dem Kanal p sendet. Kommt ein Signal (Nachricht mit leerem Tupel von Namen) auf dem Kanal e an, so erzeugt der Server einen vollkommen neuen Kanal und verschickt dessen Namen auf dem Kanal p . Kommt dagegen ein Signal auf dem Kanal z an, so verschickt der Server den Namen e über den Kanal p .

Wie im Fall von F' , sollen diese Dienste beliebig lange bereitstehen.

Aufgabe 22 *Typen von Prozessen*

Bestimmen Sie jeweils eine Funktion ob und eine Typumgebung Γ , so dass die Prozesse aus Aufgabe 20 wohlgetypt sind. Geben Sie die Ableitung des Typs explizit an.

Aufgabe 23 *Prozess mit Laufzeitfehlern*

Argumentieren Sie mit Hilfe der Typregeln, warum dem Prozess

$$P = \bar{x}\langle y \rangle.\mathbf{0} \mid y(w).\mathbf{0} \mid x(z).\bar{z}\langle z_1, z_2 \rangle.\mathbf{0}$$

kein Typ zugewiesen werden kann.

Aufgabe 24 *Prozess ohne Laufzeitfehler*

Bestimmen Sie einen Prozess, der keinen Laufzeitfehler erzeugt, aber dennoch nicht getypt werden kann.