RHEINISCH-
WESTFÄLISCHE
TECHNISCHE
HOCHSCHULE
AACHEN

LEHR- UND FORSCHUNGSGEBIET INFORMATIK II
RWTH Aachen · D-52056 Aachen · GERMANY
http://www-i2.informatik.rwth-aachen.de/lufgi2.html

Jürgen Giesl
Darius Dlugosz

Automatisierte Programmverifikation

Übungsblatt 4

Alle Aufgaben werden korrigiert. Die Abgabe ist bis Dienstag, 12. Dezember 2000, in der Übung, Vorlesung oder im Raum 4211 möglich. Die Aufgaben werden in der Übung am 12. Dezember 2000 besprochen.

Aufgabe 1: (2 Punkte) Sei P ein terminierendes Programm. Beweisen Sie: Für jede definierende Gleichung $f(t^*) \equiv r$ mit den Variablen x^* der Sorten w gilt: $M_P \models \forall x^* : w \;\; f(t^*) \equiv r$.

Aufgabe 2: (2 Punkte) Das Programm P bestehe aus den folgenden Datenstrukturen und Algorithmen:

structure number

$\mathcal{O}$	:	number
succ	:	number $\rightarrow$ number

function plus : number $\times$ number $\rightarrow$ number

plus$(\mathcal{O}, y)$	$\equiv$	y
plus(succ$(x), y)$	$\equiv$	succ(plus(x, y))

Beweisen Sie durch strukturelle Induktion über $q_1 \in \mathcal{T}(\Sigma^c)_{\mathsf{number}}$:

Für alle $q_1, q_2, q_3 \in \mathcal{T}(\Sigma^c)_{\mathsf{number}}$ gilt $\mathsf{plus}(q_1, \mathsf{plus}(q_2, q_3)) \equiv \mathsf{plus}(\mathsf{plus}(q_1, q_2), q_3) \in Th_P$.

Aufgabe 3: (0.5 Punkte) Das Programm P aus Aufgabe 2 sei um die folgenden Datenstrukturen erweitert:

$$
\begin{array}{l}
\textbf{structure}\ \mathsf{sexpr} \\
\quad \mathsf{nil} \quad : \quad \mathsf{sexpr} \\
\quad \mathsf{atom} \quad : \quad \mathsf{number} \to \mathsf{sexpr} \\
\quad \mathsf{cons} \quad : \quad \mathsf{sexpr} \times \mathsf{sexpr} \to \mathsf{sexpr} \\
\\
\textbf{structure}\ \mathsf{tree} \\
\quad \mathsf{tip} \quad : \quad \mathsf{tree} \\
\quad \mathsf{node} \quad : \quad \mathsf{tree} \times \mathsf{number} \times \mathsf{tree} \to \mathsf{tree}
\end{array}
$$

Geben Sie die Axiome AX_{sexpr} und AX_{tree} an.

Aufgabe 4: (2 Punkte) Das Programm P bestehe aus der Datenstruktur number und den folgenden Algorithmen:

$$
\begin{array}{l}
\textbf{function}\ \mathsf{double} : \mathsf{number} \to \mathsf{number} \\
\quad \mathsf{double}(\mathcal{O}) \equiv \mathcal{O} \\
\quad \mathsf{double}(\mathsf{succ}(x)) \equiv \mathsf{succ}(\mathsf{succ}(\mathsf{double}(x))) \\
\\
\textbf{function}\ \mathsf{half} : \mathsf{number} \to \mathsf{number} \\
\quad \mathsf{half}(\mathcal{O}) \equiv \mathcal{O} \\
\quad \mathsf{half}(\mathsf{succ}(\mathcal{O})) \equiv \mathcal{O} \\
\quad \mathsf{half}(\mathsf{succ}(\mathsf{succ}(x))) \equiv \mathsf{succ}(\mathsf{half}(x))
\end{array}
$$

Sei AX_P die Menge der allquantifizierten definierenden Gleichungen E_P, d.h $AX_P = \{\forall x^* : w \quad l \equiv r \mid l \equiv r \in E_P$ mit den Variablen x^* der Sorten $w\}$. Zeigen Sie: $AX_P \not\models \forall x : \mathsf{number}\ \ x \equiv \mathsf{half}(\mathsf{double}(x))$

Hinweis: Geben Sie eine Interpretation $I = (\mathcal{A}, \alpha)$ an, die ein Modell von AX_P und kein Modell von $\forall x : \mathsf{number}\ \ x \equiv \mathsf{half}(\mathsf{double}(x))$ ist. Eine Möglichkeit ist hierbei $\mathcal{A}_{\mathsf{number}} = \mathbb{N} \cup \{\top, \bot\}$ für zwei zusätzliche Objekte $\top, \bot$ zu wählen.