

Programowanie rozszerzone

Lista zadań nr 6

Na ćwiczenia 13 i 14 kwietnia 2010

Zadanie 1 (2 pkt). Rozważmy program `Euclid` napisany w języku IMP:

`while  $\neg(m = n)$  do if  $m \leq n$  then  $n := n - m$  else  $m := m - n$`

Udowodnij, że $\forall \sigma \in \Sigma. \sigma(m) \geq 1 \wedge \sigma(n) \geq 1 \Rightarrow \exists \sigma'. \langle \text{Euclid}, \sigma \rangle \rightarrow \sigma'$. Wskazówka: zdefiniuj odpowiednią relację dobrze ufundowaną na pewnym podzbiorze Σ , a następnie posłuż się indukcją noetherowską.

Zadanie 2 (2 pkt). Sprawdź, które z poniższych instrukcji są semantycznie równoważne (podaj dowód albo kontrprzykład):

1. `if  $b$  then  $(c; c_1)$  else  $(c; c_2)$` oraz `$c; \text{if } b \text{ then } c_1 \text{ else } c_2$`
2. `if  $b$  then  $(c_1; c)$  else  $(c_2; c)$` oraz `$(\text{if } b \text{ then } c_1 \text{ else } c_2); c$`
3. `repeat  $c$  until  $b$` oraz `$c; \text{while } \neg b \text{ do } c$` .

Zadanie 3 (2 pkt). Przypomnijmy składnię abstrakcyjną języka IMP:

$a ::= n \mid x \mid a_0 + a_1$
 $b ::= \text{true} \mid \text{false} \mid a_0 \leq a_1 \mid \neg b \mid b_0 \wedge b_1$
 $c ::= \text{skip} \mid x := a \mid c_0; c_1 \mid \text{if } b \text{ then } c_0 \text{ else } c_1 \mid \text{while } b \text{ do } c$

Do tej pory rozważaliśmy strukturalną semantykę operacyjną dużych kroków (semantykę naturalną) języka IMP. W tym zadaniu zajmiemy się strukturalną semantyką operacyjną małych kroków dla języka IMP. Oto indukcyjna definicja relacji $\Rightarrow \subseteq \mathbf{Aexp} \times \Sigma \times \mathbf{Aexp} \times \Sigma$, opisującej pojedynczy krok ewaluacji wyrażeń arytmetycznych:

$$\frac{\langle a_0, \sigma \rangle \Rightarrow \langle a'_0, \sigma' \rangle}{\langle a_0 + a_1, \sigma \rangle \Rightarrow \langle a'_0 + a_1, \sigma' \rangle} \quad \frac{\langle a_1, \sigma \rangle \Rightarrow \langle a'_1, \sigma' \rangle}{\langle n_0 + a_1, \sigma \rangle \Rightarrow \langle n_0 + a'_1, \sigma' \rangle}$$
$$\overline{\langle n_0 + n_1, \sigma \rangle \Rightarrow \langle n, \sigma \rangle}^n = n_0 + n_1 \quad \overline{\langle x, \sigma \rangle \Rightarrow \langle \sigma(x), \sigma \rangle}$$

1. Zaproponuj reguły definiujące strukturalną semantykę operacyjną małych kroków dla wyrażeń logicznych ($\Rightarrow \subseteq \mathbf{Bexp} \times \Sigma \times \mathbf{Bexp} \times \Sigma$) i instrukcji ($\Rightarrow \subseteq \mathbf{Com} \times \Sigma \times ((\mathbf{Com} \times \Sigma) \cup \Sigma)$) języka IMP.
2. Dla chętnych (nie musisz umieć wykonać tego polecenia, żeby zadeklarować zadanie): udowodnij, że semantyka dużych kroków i semantyka małych kroków języka IMP są równoważne, tzn. $\langle c, \sigma \rangle \rightarrow \sigma'$ wtedy i tylko wtedy, gdy $\langle c, \sigma \rangle \Rightarrow^* \sigma'$.
3. Zadać semantykę małych kroków dla instrukcji $c_0 || c_1$ (równoległe wykonanie c_0 i c_1) i `abort` (awaryjne przerwanie wykonania programu).