

Programowanie (M) 2013

Lista zadań nr 6

Na ćwiczenia 22 maja 2013

Zadanie 1 (1 pkt). Niech $\mathcal{F} = \langle \mathbb{N} \hookrightarrow \mathbb{N}, \subseteq \rangle$ będzie zbiorem wszystkich funkcji częściowych z $\mathbb{N}$ w $\mathbb{N}$ z porządkiem wyznaczonym przez inkluzję. Udowodnij, że istnieje najmniejsza funkcja $g \in \mathcal{F}$ spełniająca równanie $g = \Phi g$, gdzie

$$\Phi(f)(n) = \begin{cases} 1, & \text{gdy } n = 0 \\ n \cdot f(n-1), & \text{gdy } n > 0 \text{ i } n-1 \in \text{Dom}(f) \\ \text{niezdefiniowana}, & \text{w p.p.} \end{cases}$$

Wyznacz tę funkcję. Jak wyglądają pozostałe punkty stałe funkcji Φ ?

Zadanie 2 (1 pkt). Udowodnij, że definicja semantyki denotacyjnej języka **While** jest poprawna, tzn., że $\mathcal{C}$ jest totalną funkcją z Stm w $State \hookrightarrow State$.

Zadanie 3 (1 pkt). Udowodnij równoważność semantyki naturalnej oraz denotacyjnej języka **While**, tzn. pokaż, że

$$\langle S, s \rangle \rightarrow s' \text{ wtedy i tylko wtedy, gdy } \mathcal{C}\llbracket S \rrbracket s = s'.$$

Zadanie 4 (1 pkt). Wyznacz semantykę denotacyjną instrukcji

$$z := 0; \text{while } y \leq x \text{ do } (z := z + 1; x := x - y).$$

Zadanie 5 (1 pkt). Wyznacz semantykę denotacyjną instrukcji

$$x := 0; \text{while } 0 \leq x \text{ do } x := x + 1.$$

Zadanie 6 (1 pkt). Rozszerz definicję semantyki denotacyjnej języka **While** o równanie definiujące semantykę instrukcji `repeat  $S$  until  $b$` . Uzasadnij poprawność swojej definicji.

Zadanie 7 (1 pkt). Wyprowadź równanie semantyczne dla instrukcji iteracji `for  $x := a_1$  to  $a_2$  do  $S$` , traktując ją jako cukier syntaktyczny w języku **While**.

Zadanie 8 (1 pkt). Rozważmy zbiór $\mathbb{L}$ zawierający listy (skończone, częściowe i nieskończone) liczb całkowitych, wraz z relacją $\leq$ zdefiniowaną indukcyjnie następująco:

$$\begin{aligned} \perp &\leq xs, \\ x : xs &\leq x : xs' \text{ jeżeli } xs \leq xs'. \end{aligned}$$

Pokaż, że $(\mathbb{L}, \leq)$ jest zupełnym porządkiem częściowym. Wyznacz najmniejsze górne ograniczenie łańcucha $\{\perp, 1 : \perp, 1 : 2 : \perp, 1 : 2 : 3 : \perp, \dots\}$.

Niech $\mathbb{T}$ będzie zbiorem $\{\text{tt}, \text{ff}, \perp\}$ z porządkiem zdefiniowanym następująco:

$$x \leq y \text{ gdy } x = y \vee x = \perp.$$

Czy funkcja $f : \mathbb{L} \rightarrow \mathbb{T}$, zdefiniowana następująco:

$$f(xs) = \begin{cases} \text{tt}, & 42 \text{ występuje w } xs \\ \text{ff}, & 42 \text{ nie występuje w } xs \text{ i } xs \text{ nie jest listą częściową} \\ \perp, & 42 \text{ nie występuje w } xs \text{ i } xs \text{ jest listą częściową} \end{cases}$$

jest ciągła? Czy można ją zaimplementować w Haskellu? Jaką funkcję z $\mathbb{L}$ w $\mathbb{T}$ implementuje następujący fragment kodu w Haskellu i czy jest ona ciągła?

```
find42 :: [Int] -> Bool
find42 [] = False
find42 (x:xs) = if x == 42
                  then True
                  else find42 xs
```