

# Semantyka języków programowania

II UW r 2013/14

## Lista zadań nr 2

Na ćwiczenia 22 października 2013

**Zadanie 1.** Zdefiniuj indukcyjnie funkcję  $\text{loc}_a : \mathbf{Aexp} \rightarrow \mathcal{P}(\mathbf{Loc})$ , która dla danego wyrażenia arytmetycznego wyznacza zbiór wszystkich zmiennych występujących w tym wyrażeniu. Pokaż następnie, że wartość wyrażenia arytmetycznego zależy wyłącznie od wartości zmiennych w nim występujących, tzn. pokaż, że jeżeli  $\sigma(x) = \sigma'(x)$  dla wszystkich  $x \in \text{loc}_a(a)$ , to  $\langle a, \sigma \rangle \rightarrow n$  wtedy i tylko wtedy, gdy  $\langle a, \sigma' \rangle \rightarrow n$ .

Udowodnij podobne twierdzenie dla wyrażeń logicznych. A jak mogłoby wyglądać takie twierdzenie dla instrukcji?

**Zadanie 2.** Dany jest następujący program  $P$  w języku IMP:

$$z := 1; \text{while } \neg(x = 0) \text{ do } (z := z * y; x := x - 1)$$

Pokaż, że jeżeli  $\langle P, \sigma \rangle \rightarrow \sigma'$ , to  $\sigma'(z) = \sigma(y)^{\sigma(x)}$ .

**Zadanie 3.** Udowodnij, że jeżeli  $\sigma(x) < 0$  oraz  $x \notin \text{loc}_L(c)$  (gdzie  $\text{loc}_L(c)$  jest zbiorem zmiennych, które występują po lewej stronie instrukcji przypisania w  $c$ ), to nie istnieje  $\sigma'$  takie, że

$$\langle \text{while } \neg(x = 0) \text{ do } (c; x := x - 1), \sigma \rangle \rightarrow \sigma'.$$

**Zadanie 4.** Niech  $W \equiv \text{while } b \text{ do } c$ . Udowodnij, że  $\langle W, \sigma \rangle \rightarrow \sigma'$  wtedy i tylko wtedy, gdy  $\langle b, \sigma \rangle \rightarrow \text{false} \wedge \sigma = \sigma'$  lub  $\exists \sigma_0, \dots, \sigma_n. \sigma = \sigma_0 \wedge \sigma' = \sigma_n \wedge \langle b, \sigma_n \rangle \rightarrow \text{false} \wedge (\forall 0 \leq i < n. \langle b, \sigma_i \rangle \rightarrow \text{true} \wedge \langle c, \sigma_i \rangle \rightarrow \sigma_{i+1})$ .

**Zadanie 5.** Zadaj semantykę naturalną instrukcji

$$\text{repeat } c \text{ until } b$$

(o intuicyjnej semantyce: powtarzaj instrukcję  $c$  aż warunek  $b$  będzie spełniony). Otrzymane reguły nie mogą odwoływać się do instrukcji `while`. Udowodnij, że instrukcje `repeat c until b` oraz `c; while  $\neg b$  do c` są semantycznie równoważne.