

## SEMANTYKA I WERYFIKACJA - Zadanie domowe nr 1

Napisz semantykę operacyjną dużych kroków instrukcji języka o następującej gramatyce:

```
Num  $\ni n ::= 0 \mid 1 \mid -1 \mid 2 \mid -2 \mid \dots$   
Var  $\ni x ::= x \mid y \mid \dots$   
Expr  $\ni e ::= n \mid x \mid e_1 + e_2 \mid e_1 * e_2 \mid e_1 - e_2$   
BExpr  $\ni b ::= \text{true} \mid \text{false} \mid e_1 < e_2 \mid e_1 = e_2 \mid b_1 \wedge b_2 \mid \neg b$   
Instr  $\ni I ::= \text{skip} \mid x := e \mid I_1; I_2 \mid \text{if } b \text{ then } I_1 \text{ else } I_2 \mid \text{while } b \text{ do } I \mid$   
           $\text{limit } e \text{ in } I \mid \text{allow } e \text{ in } I$ 
```

Wyliczanie wyrażeń arytmetycznych i logicznych odbywa się standardowo. Można założyć, że semantyka wyrażeń jest dana.

Wykonanie każdego przypisania  $x := e$  wiąże się z kosztem równym wartości bezwzględnej przypisywanej liczby. Instrukcja

`limit e in I`

rozpoczyna się od obliczenia wartości wyrażenia  $e$ . Jeżeli ta wartość jest nieujemna, to następnie wykonuje się instrukcja  $I$ , z tym zastrzeżeniem, że sumaryczny koszt wykonanych w niej przypisań nie może przekroczyć  $e$ . Jeżeli takie przekroczenie nastąpi, to wszystkie efekty przypisań wykonanych w instrukcji  $I$  są odwoływane, ale ich koszt jest ponoszony, tj. koszt instrukcji `limit e in I` jest równy kosztowi wszystkich wykonanych w niej przypisań wraz z tym, które spowodowało przekroczenie limitu.

Instrukcja

`allow e in I`

wykonuje się podobnie jak `limit . . in . .`, jednak z tą różnicą, że w instrukcji  $I$  ignorowane są wszystkie limity narzucone przez ew. otaczające instrukcje `limit` i `allow`. Jeżeli wykonanie  $I$  zakończy się sukcesem to koszt instrukcji `allow e in I` wynosi 0, w przeciwnym razie jest równy różnicy między sumarycznym kosztem wykonanej części instrukcji  $I$  a obliczonym na wstępie limitem  $e$ .

Przykładowo, po wykonaniu ciągu instrukcji

```
x:=0;  
limit 2 in  
  allow 4 in  
    x := 5;  
    x := x+1;  
limit 5 in  
  limit 100 in  
    x := x+5;  
limit 100 in  
  while true do x:=3;
```

(gdzie wcięcia określają zasięg instrukcji `limit` i `allow`) zmienna  $x$  przyjmuje wartość 1.