Розширення мови SIPL масивами

1 Синтаксис розширення

Ліва частина правила	Права частина правила	Ім'я правила
<програма> ::=	**begin** <оператор> **end**	NP1
<оператор> ::=	<змінна> [<вираз>] := <вираз> <змінна> := <вираз> <оператор> ; <оператор> **if** <умова> **then** <оператор> **else** <оператор> **while** <умова> **do** <оператор> **begin** <оператор> **end** **skip**	NS1 NS2 NS3 NS4 NS5 NS6 NS7
<вираз> ::=	<змінна> [<вираз>] | <число> | <змінна> | <вираз> + <вираз> | <вираз> − <вираз> | <вираз> ∗ <вираз> | (<вираз>)	NA1 . . . . . . NA7
<умова> ::=	<вираз> = <вираз> | <вираз> > <вираз> | <умова> ∨ <умова> | ¬ <умова> | (<умова>)	NB1 . . . NB5
<змінна> ::=	. . . | M | N | . . .	NV. . .

<число> ::=	...| -1 | 0 | 1 | 2 | ...	NN...

2 Композиційна семантика розширення

2.1 Дані

1. $Int = \{\dots, -1, 0, 1, 2, \dots\}Var$;
2. $Bool = \{true, false\}$
3. $Var = \{\dots, M, N, \dots\}$
4. $Arr\_Var = \bigcup_{i=-\infty}^{\infty} \{\dots, M.i, N.i, \dots\}$;
5. $State = Var \cup Arr\_Var \to Int.$

2.2 Алгебри

1. Алгебра цілих чисел: A_Int = <Int; add, sub, mult>
2. Алгебра булевих значень: A_Bool = <Bool; or, neg>
3. Алгебра базових даних: A_Int_Bool = <Int, Bool; add, sub, mult, or, neg, eq, gr>
4. Багатоосновна алгебра мови SIPL: A_Int_Bool_State = <Int, Bool, State; add, sub, mult, or, neg, eq, gr, ⇒x, x⇒, n , id, ∇>

2.3 Функції

1. n-арні операції над базовими типами:
   - $FNA = Int^n \to Int$ – n-арні арифметичні функції (операції);
   - $FNB = Bool^n \to Bool$ – n-арні булеві функції (операції);
   - $FNAB = Int^n \to Bool$ – n-арні функції (операції) порівняння.
2. Функції над станами змінних:

- $FA = State \to Int$ – номінативні арифметичні функції;
- $FB = State \to Bool$ – номінативні предикати;
- $FS = State \to State$ – біномінативні функції-перетворювачі. (трансформатори) станів

2.4 Композиції

1. Композиції, які пов'язані з номінативними функціями та предикатами:

   - $S^n : FNA \times FA^n \to FA$ – суперпозиція номінативних арифметичних функцій у n-арну арифметичну функцію;
   - $S^n : FNAB \times FA^n \to FB$ – суперпозиція номінативних арифметичних функцій у n-арну функцію порівняння
   - $S^n : FNB \times FB^n \to FB$ – суперпозиція номінативних предикатів у n-арну булеву функцію

2. Композиції, які пов'язані з біномінативними функціями:

   - $AS^x : FA \to FS$ – оператор присвоювання для змінних, який задається формулою:

   $$AS^x(fa)(st) = st\nabla[x \mapsto fa(st)]$$

   - $AS^x : FA \times FA \to FS$ – оператор присвоювання для комірок масивів, який задається формулою:

   $$AS^x(ga,fa)(st) = st\nabla[x.v \mapsto fa(st)],\ v = ga(st)$$

   - $FS^2 \to FS$ – оператор послідовного виконання, який задається формулою:

   $$(fs_1 \circ fs_2)(st) = fs_2(fs_1(st))$$

   - $IF : FB \times FS^2 \to FS$ – умовний оператор, який задається формулою:

   $$IF(fb, fs_1, fs_2)(st) = \begin{cases} fs_1(st), & \text{якщо } fb(st) = true \\ fs_2(st), & \text{якщо } fb(st) = false \end{cases}$$

- $WH : FB \times FS \to FS$ – оператор циклу, який задається індуктивно:

$$WH(fb, fs)(st) = st_n,$$

де $st_0 = st$, $st_1 = fs(st_0)$, $\ldots$, $st_n = fs(st_{n-1})$,
причому $fb(st_0) = true, \ldots fb(st_{n-1}) = true, fb(st_n) = false.$

- $VAL^x \ : FA \to FA$ – оператор розіменування

$$VAL^x \ (fa)(st) = x.v \Rightarrow (st), v = fa(st)$$

3 Правила перетворення програми на семантичний терм

sem_P(begin S end) = sem_S(S)
sem_S(x := a) = AS^x (sem_A(a))
sem_s(x[b] := a) = AS^x (sem_A(b), sem_A(a))
sem_S(S1; S2) = sem_S(S1) $\circ$ sem_S(S2)
sem_S(if b then S1 else S2) = IF(sem_B(b), sem_S(S1), sem_S(S2))
sem_S(while b do S) = WH(sem_B(b), sem_S(S))
sem_S(begin S end) = sem_S(S)
sem_S(skip) = id
sem_A(n) = ($\overline{n}$)
sem_A(x) = x$\Rightarrow$
sem_A(x[a]) = VAL^x (sem_A(a))
sem_A(a + b) = S^2 (add, sem_A(a), sem_A(b))
sem_A(a - b) = S^2 (sub, sem_A(a), sem_A(b))
sem_A(a * b) = S^2 (mult, sem_A(a), sem_A(b))
sem_A((a)) = sem_A(a)
sem_B(a = b) = S^2 (eq, sem_A(a), sem_A(b))
sem_B(a > b) = S^2 (gr, sem_A(a), sem_B(b))
sem_B(a *lor* b) = S^2 (or, sem_B(a), sem_B(b))
sem_B($lnot$a) = S^1 (neg, sem_B(a))

$$\boxed{\text{sem\_B}((\text{b})) = \text{sem\_B}(\text{b})}$$

4 Операційна семантика розширення

Назва правила	Правило операційної семантики
	Правила для програми та операторів
PR	$\dfrac{< S, st >\to st'}{< begin\ S\ end,\ st >\to st'}$
AS	$\dfrac{< a,\ st >\to n}{< x := a,\ st >\to st\nabla[x \to n]}$
<u>ASM</u>	$\dfrac{< a,\ st >\mapsto n_1 \qquad < i,\ st >\mapsto n_2}{< X[i] := a,\ st >\mapsto st\nabla[X.n_2 \mapsto n_1]}$
SEQ	$\dfrac{< S_1,\ st >\to st_1 \qquad < S_2,\ st_1 >\to st_2}{< S_1;\ S_2,\ st >\to st_2}$
IFtrue	$\dfrac{< b,\ st_1 >\to true \qquad < S_1,\ st_1 >\to st_2}{< if\ b\ then\ S_1\ else\ S_2,\ st_1 >\to st_2}$
IFfalse	$\dfrac{< b,\ st_1 >\to false \qquad < S_2,\ st_1 >\to st_2}{< if\ b\ then\ S_1\ else\ S_2,\ st_1 >\to st_2}$
WHfalse	$\dfrac{< b,\ st >\to false}{< while\ b\ do\ S,\ st >\to st}$
WHtrue	$\dfrac{< b,\ st >\to true \quad < S,\ st >\to st_1 \quad < while\ b\ do\ S,\ st_1 >\to st_2}{< while\ b\ do\ S,\ st >\to st_2}$
BEG	$\dfrac{< S,\ st_1 >\to st_2}{< begin\ S\ end,\ st_1 >\to st_2}$
skip	$< skip,\ st >\to st$
	Правила для числових виразів
Num	$< n,\ st >\to n$
Var	$< x,\ st >\to st(x)$
<u>VarM</u>	$\dfrac{< i, st >\mapsto n}{< X[i], st >\mapsto st(X.n)}$
A+	$\dfrac{< a,\ st >\to n \qquad < b,\ st >\to m}{< a + b,\ st >\to add(n, m)}$
A-	$\dfrac{< a,\ st >\to n \qquad < b,\ st >\to m}{< a - b,\ st >\to sum(n, m)}$

A*	$\dfrac{< a,\, st > \to n \qquad\qquad < b,\, st > \to m}{< a * b,\, st > \to mult(n, m)}$
A()	$\dfrac{< a,\, st > \to st}{< (a),\, st > \to st}$
Правила для умов	
B=	$\dfrac{< a,\, st > \to n \qquad\qquad < b,\, st > \to m}{< a = b,\, st > \to eq(n, m)}$
B>	$\dfrac{< a,\, st > \to n \qquad\qquad < b,\, st > \to m}{< a > b,\, st > \to gr(n, m)}$
B∨	$\dfrac{< a,\, st > \to r_1 \qquad\qquad < b,\, st > \to r_2}{< a \vee b,\, st > \to or(r_1, r_2)}$
B¬	$\dfrac{< a,\, st > \to r_1}{< \neg a,\, st > \to neg(r_1)}$
B()	$\dfrac{< b,\, st > \to r}{< (b),\, st > \to r}$

5 Аксіоматична семантика розширення

Правило виведення	**Позначення правила**
$\{P[x \mapsto a]\}\ x := a\ \{P\}$	AS
$\{P[X.n \mapsto a]\}\ X[N] := a\ \{P\}$	ASM
$\{P\}\ skip\ \{P\}$	skip
$\dfrac{\{P\}\ S_1\ \{Q\},\{Q\}\ S_2\ \{R\}}{\{P\}\ S_1; S_2\ \{R\}}$	S
$\dfrac{\{b \land P\}\ S_1\ \{Q\},\ \{\neg b \land P\}\ S_2\ \{Q\}}{\{P\}\ \boldsymbol{if}\ b\ \boldsymbol{then}\ S_1\ \boldsymbol{else}\ S_2\ \{Q\}}$	IF
$\dfrac{\{b \land P\}\ S\ \{P\}}{\{P\}\ \boldsymbol{while}\ b\ \boldsymbol{do}\ S\{\neg b \land P\}}$	WH
$\dfrac{\{P'\}\ S\ \{Q'\}}{\{P\}\ S\ \{Q\}}$, якщо $P \Rightarrow P'$ та $Q' \Rightarrow Q$	C
$\dfrac{\{P\}\ S\ \{Q\}}{\{P\}\ \boldsymbol{begin}\ S\ \boldsymbol{end}\ \{Q\}}$	BE

Табл. 4: Аксіоматична семантика

6 Текст програми

Вхідними даними для програми є число $N \geq 0$ та значення комірок масиву $X.i$, $0 \leq i < N$.

Введемо наступні позначення:

- L2 – I := 0
- L3 – S := 0
- L2_8 – рядки 2-8
- L4_8 – рядки 4-8
- L5_8 – рядки 5-8

Алгоритм 1: Програма ARR-SUM

```
1 begin
2 I := 0;
3 Z := 0;
4 while N > I do
5 |  begin
6 |  Z := Z + X[I];
7 |  I := I + 1
8 |  end
9 end
```

- L6_7 – рядки 6-7
- L6 – S := S + X[I]
- L7 – I := I + 1

7 Тестування програми в композиційній семантиці

7.1 Побудова семантичного терму

Sem_P(ARR_SUM)
= Sem_P(begin L2;L3;L4_8 end)
= Sem_S(L2;L3;L4_8)
= Sem_S(L2) ∘ Sem_S(L3) ∘ Sem_S(L4_8)

Sem_S(L2) = Sem_S(I := 0) = AS^I (Sem_A(0)) = AS^I ($\overline{0}$)
Sem_S(L3) = Sem_S(Z := 0) = AS^Z (Sem_A(0)) = AS^Z ($\overline{0}$)
Sem_S(L4_8)
= Sem_S(while N > I do L5_8)
= WH(Sem_B(N > I), Sem_S(begin L6_7 end))
= WH(S^2 (gr, Sem_A(N), Sem_A(I)), Sem_S(L6 ; L7))
= WH(S^2 (gr, N⇒, I⇒), Sem_S(L6)∘ Sem_S(L7))

Sem_S(L6)
= Sem_S(Z := Z + X[I])
= AS^S (Sem_A(Z + X[I]))
= AS^S (S^2 (add, Sem_A(Z), Sem_A(X[I])))
= AS^S (S^2 (add, Z⇒, VAL^X (Sem_A(I))))
= AS^S (S^2 (add, Z⇒, VAL^X (I⇒)))

Sem_S(L7)
= Sem_S(I := I + 1)
= AS^I (Sem_A(I + 1))
= AS^I (S^2 (add, Sem_A(I), Sem_A(1)))
= AS^I (S^2 (add, I⇒, $\overline{1}$))

Sem_P(ARR_SUM)
= AS^I ($\overline{0}$) ∘ AS^Z ($\overline{0}$) ∘ WH(S^2 (gr, N⇒, I⇒), AS^S (S^2 (add, Z⇒, VAL^X (I⇒)))
∘ AS^I (S^2 (add, I⇒, $\overline{1}$)))

7.2 Обчислення семантичного терму

Обчислимо побудований семантичний терм на стані

$$st = st_0 = [N \to 2, X.0 = 3, X.1 = 4]$$

$$AS^I\ (\overline{0})(st_0) = st_0 \nabla [I \to 0] = st_1$$
$$AS^Z\ (\overline{0})(st_1) = st_1 \nabla [Z \to 0] = st_2$$
$$S^2\ (gr, N \Rightarrow, I \Rightarrow)(st_2) = gr(N \Rightarrow (st_2), I \Rightarrow (st_2)) = gr(2, 0) = true$$

Умова циклу виконується, тому обчислюємо тіло циклу:

$$
\begin{aligned}
&AS^S\ (S^2\ (add, S \Rightarrow, VAL^X\ (I \Rightarrow)))(st_2) \\
&\qquad = st_2\nabla[Z \to S^2\ (add, Z \Rightarrow, VAL^X\ (I \Rightarrow))(st_2)] \\
&\qquad = st_2\nabla[Z \to add(Z \Rightarrow (st_2), VAL^X\ (I \Rightarrow (st_2)))] \\
&\qquad = st_2\nabla[Z \to add(0, VAL^X\ (0))] \\
&\qquad = st_2\nabla[Z \to add(0, 3)] \\
&\qquad = st_2\nabla[Z \to 3] = st_3
\end{aligned}
$$

$$
\begin{aligned}
&AS^I\ (S^2\ (add, I \Rightarrow, \overline{1}))(st_3) = st_3\nabla[I \to S^2\ (add, I \Rightarrow, \overline{1})(st_3)] \\
&\qquad = st_3\nabla[I \to add(I \Rightarrow (st_3), \overline{1}(st_3))] = st_3\nabla[I \to add(0, 1)] \\
&\qquad = st_3\nabla[I \to 1] = st_4
\end{aligned}
$$

Обчислюємо умову циклу:

$$S^2\ (gr, N \Rightarrow, I \Rightarrow)(st_4) = gr(N \Rightarrow (st_4), I \Rightarrow (st_4)) = gr(2, 1) = true$$

Умова циклу виконується, тому обчислюємо тіло циклу:

$$
\begin{aligned}
&AS^S\ (S^2\ (add, S \Rightarrow, VAL^X\ (I \Rightarrow)))(st_4) \\
&\qquad = st_4\nabla[Z \to S^2\ (add, Z \Rightarrow, VAL^X\ (I \Rightarrow))(st_4)] \\
&\qquad = st_4\nabla[Z \to add(Z \Rightarrow (st_4), VAL^X\ (I \Rightarrow (st_4)))] \\
&\qquad = st_4\nabla[Z \to add(3, VAL^X\ (1))] \\
&\qquad = st_4\nabla[Z \to add(3, 4)] \\
&\qquad = st_4\nabla[Z \to 7] = st_5
\end{aligned}
$$

$$
\begin{aligned}
&AS^I\ (S^2\ (add, I \Rightarrow, \overline{1}))(st_5) = st_5\nabla[I \to S^2\ (add, I \Rightarrow, \overline{1})(st_5)] \\
&\qquad = st_5\nabla[I \to add(I \Rightarrow (st_5), \overline{1}(st_5))] = st_5\nabla[I \to add(1, 1)] \\
&\qquad = st_5\nabla[I \to 1] = st_6
\end{aligned}
$$

Обчислюємо умову циклу:

$$S^2\ (gr, N\Rightarrow, I\Rightarrow)(st_5) = gr(N\Rightarrow(st_5), I\Rightarrow(st_5)) = gr(2,2) = false$$

Умова циклу не виконується, тому маємо остаточний стан:

$$st_6 = [N \to 2, X.0 \to 3, X.1 \to 4, I \to 2, Z \to 7] \tag{1}$$

8 Доведення коректності програми

8.1 Часткова коректність

Введемо позначення для термів програми:

$$\begin{aligned}
&f_1 = AS^I\ (\overline{0})\\
&f_2 = AS^Z\ (\overline{0})\\
&f_3 = WH(f_4, f_5 \circ\ f_6)\\
&f_4 = S^2\ (gr, N\Rightarrow, I\Rightarrow)\\
&f_5 = AS^S\ (S^2\ (add, Z\Rightarrow, VAL^X\ (I\Rightarrow)))\\
&f_6 = AS^I\ (S^2\ (add, I\Rightarrow, \overline{1}))
\end{aligned}$$

Обчислимо програму на стані $[N \to n, X.0 \to x_0, \dots, X.n-1 \to x_{n-1}]$:

$$(f_1 \circ\ f_2 \circ\ f_3)(st) = f_3(f_2(f_1(st))) = f_3(f_2(st\nabla[I \to 0])) = f_3(st\nabla[I \to 0, Z \to 0])$$

Потрібно довести, що якщо значення $f_3(st')$ є визначеним, то воно дорівнює $st'' = st'\nabla[I \to n, Z \to \sum_{i=0}^{n-1} X.i]$. Стан $st' = st\nabla[I \to 0, Z \to 0]$

Припустимо, що за k ітерацій циклу стан має вигляд $st_k = st\nabla[I \to k, Z \to \sum_{i=0}^{k-1} X.i]$. Доведемо гіпотезу за індукцією по k.

<u>База індукції</u> для $k = 0$. Цикл не виконувався жодного разу, тому $st_0 = st'$. Гіпотезу доведено для $k = 0$.

<u>Крок індукції</u>. Припустимо, гіпотеза виконується для $k = t$, доведемо, що за $t+1$ ітерацій циклу стан буде дорівнювати $st_{t+1} = st\nabla[I \to t+1, Z \to \sum_{i=0}^{t} X.i]$.

Обчислимо значення умови на стані $st_t = st\nabla[I \to t, Z \to \sum_{i=0}^{t-1} X.i]$:

$$f_4(st_t) = S^2\ (gr, N\Rightarrow, I\Rightarrow)(st_t) = gr(N\Rightarrow(st_t), I\Rightarrow(st_t)) = gr(n,t)$$

Оскільки цикл виконується хоча б $t+1$ разів, то значення умови дорівнює $true$. Обчислимо $f_5 \circ\ f_6(st_t)$:

$$
\begin{aligned}
f_5(st_t) = AS^S\ (S^2\ (add, Z \Rightarrow, VAL^X\ (I \Rightarrow)))(st_t) \\
= st_t \nabla [Z \to S^2\ (add, Z \Rightarrow, VAL^X\ (I \Rightarrow))(st_t)] \\
= st_t \nabla [Z \to add(Z \Rightarrow (st_t), VAL^X\ (I \Rightarrow)(st_t))] \\
= st_t \nabla [Z \to add(\sum_{i=0}^{t-1} X.i, VAL^X\ (I \Rightarrow (st_t)))] \\
= st_t \nabla [Z \to add(\sum_{i=0}^{t-1} X.i, VAL^X\ (t))] \\
= st_t \nabla [Z \to \sum_{i=0}^{t} X.i] = st'_t
\end{aligned}
$$

$$
\begin{aligned}
f_6(st'_t) = AS^I\ (S^2\ (add, I \Rightarrow, \overline{1}))(st'_t) = st'_t \nabla [I \to add(I \Rightarrow (st'_t), \overline{1}(st'_t))] \\
= st'_t \nabla [I \to add(t, 1)] = st'_t \nabla [I \to t+1] = st_{t+1}
\end{aligned}
$$

Гіпотеза є доведеною. Тепер неважко бачити, що умова циклу є істиною на стані st_i, $i < n$ та стає хибною на стані st_n. Отже, часткова коректність програми є доведеною.

8.2 Повна коректність

Фактично, треба показати, що при правильних вхідних даних програма є завершуваною. Ця умова є еквівалентною умові того, що кількість ітерацій циклу є скінченною, оскільки інші оператори на це не впливають.

З доведеної раніше гіпотези маємо те, що стани програми під час виконання циклу утворюють послідовність $st_0, \ \ldots, \ st_t, \ st_{t+1}, \ \ldots$. Тому значення змінної I утворюють монотонну послідовність $0, \ \ldots, \ t, \ t+1, \ \ldots$. Також застосуємо умову того, що значення змінної N є скінченним. Тому за скінченну кількість ітерацій значення змінної I стане рівним n, і умова циклу стане хибною, тобто цикл завершиться. Тому програма завжди завершується за скінченну кількість ітерацій.

Повна коректність програми є доведеною.

9 Тестування програми в операційній семантиці

Побудуємо виведення для формули <ARR-SUM, st $>\rightarrow$ st', $st = [N \rightarrow 1,\ X.0 = 10]$

$$PR: \frac{< begin\ L_{2-8}\ end,\ st > \rightarrow st'}{SEQ: \dfrac{< L_2; L_3; L_{4-8},\ st > \rightarrow st'}{< L_2,\ st > \rightarrow st_1 \qquad < L_3; L_{4-8},\ st_1 > \rightarrow st'}}$$

Обчислимо st_1:

$$AS: \frac{< 0,\ st > \rightarrow st}{< I := 0,\ st > \rightarrow st \nabla [I \rightarrow 0]}$$

Маємо $st_1 = [N \rightarrow 1,\ X.0 = 10,\ I \rightarrow 0]$

$$SEQ: \frac{< L_3; L_{4-8},\ st_1 > \rightarrow st'}{< L_3,\ st_1 > \rightarrow st_2 \qquad < L_{4-8},\ st_2 >,\ \rightarrow st'}$$

Обчислимо st_2:

$$AS: \frac{< 0,\ st_1 > \rightarrow st_1}{< Z := 0,\ st_1 > \rightarrow st_1 \nabla [Z \rightarrow 0]}$$

Маємо $st_2 = [N \rightarrow 1,\ X.0 = 10,\ I \rightarrow 0,\ Z \rightarrow 0]$

$$WHr: \frac{< while\ N > I\ do\ L_{5-8},\ st_2 > \rightarrow st'}{< N > I,\ st_2 > \rightarrow r}$$

Обчислимо r:

$$B >: \frac{< N,\ st_2 > \rightarrow 1 \qquad < I,\ st_2 > \rightarrow 0}{< N > I,\ st_2 > \rightarrow gr(1,0) = true}$$

Маємо $r = true$:

$$WHtrue: \frac{< while\ N > I\ do\ L_{5-8},\ st_2 > \rightarrow st'}{\begin{array}{r}< N > I,\ st_2 > \rightarrow true \mid\ < L_{5-8},\ st_2 > \rightarrow st_4 \mid \\ < while\ N > I\ do\ L_{5-8},\ st_4 > \rightarrow st'\end{array}}$$

Обчислимо s_4:

$$BEG: \frac{< begin\ L_{6-7}\ end,\ st_2 > \rightarrow s_4}{SEQ: \dfrac{< L_6;\ L_7,\ st_2 > \rightarrow s_4}{< L_6,\ st_2 > \rightarrow st_3 \qquad < L_7,\ st_3 > \rightarrow st_4}}$$

Обчислимо s_3:

$$VarM: \frac{< I, st_2 > \to 0}{< X[I], st_2 > \to st_2(X.0) = 10}$$

$$A+: \frac{< Z, st_2 > \to 0 \qquad < X[I], st_2 > \to 10}{< Z + X[I], st_2 > \to add(0, 10) = 10}$$

$$AS: \frac{< Z + X[I], st_2 > \to 10}{< Z := Z + X[I], st_2 > \to st_2 \nabla [Z \to 10]}$$

Маємо $st_3 = [N \to 1,\ X.0 = 10,\ I \to 0,\ Z \to 10]$, обчислимо st_4

$$A+: \frac{< I, st_3 > \to 0 \qquad < 1, st_3 > \to 1}{AS: \dfrac{< I + 1, st_3 > \to add(0, 1) = 1}{< I = I + 1, st_3 > \to st_3 \nabla [I \to 1]}}$$

Маємо $st_4 = [N \to 1,\ X.0 = 10,\ I \to 1,\ Z \to 10]$

$$WHr: \frac{< while\ N > I\ do\ L_{5-8}, st_4 > \to st'}{< N > I, st_4 > \to r}$$

Обчислимо r:

$$B >: \frac{< N, st_4 > \to 1 \qquad < I, st_4 > \to 1}{< N > I, st_2 > \to gr(1, 1) = false}$$

Маємо $r = false$, тому $st_4 = st' = [N \to 1,\ X.0 = 10,\ I \to 1,\ Z \to 10]$:

10 Доведення еквівалентності операційної та композиційної семантик

Теорема. Для довільної програми мови $SIPL$ її операційна та композиційна семантика є еквівалентними.

1. Композиційна та операційна семантики кожного числового виразу є еквівалентними (випливає з еквівалентності окремих правил обох семантик).

2. Композиційна та операційна семантики кожної умови є еквівалентними (випливає з еквівалентності окремих правил обох семантик).

3. Семантика інших функцій та операторів (операторів присвоєння для змінних, присвоєння для комірок масивів, умови, послідовного виконання і т.д.) крім оператора циклу є еквівалентними (випливає з еквівалентності окремих правил обох семантик).

4. Доводимо еквівалентність оператору циклу наступним чином:

   (а) Якщо тіло циклу виконується 0 разів, то стани програм в обох семантиках є рівними з попередньо доведеної еквівалентності функцій та операторів.

   (б) Якщо стани програм є рівними на ітерації k, то вони будуть рівними і при виконання $k+1$ ітерації з попередньо доведеної еквівалентності функцій та операторів.

   Таким чином семантика оператору циклу є еквівалентною.