

Математические модели последовательных вычислений

mk.cs.msu.ru → Лекционные курсы
→ Математические модели последовательных вычислений

Блок 14

Стандартные схемы программ

Лектор:

Подымов Владислав Васильевич

E-mail:

valdus@yandex.ru

Будем считать заданными:

1. Счётное множество **переменных** Var
2. Сигнатуру логики предикатов $\langle \text{Const}, \text{Func}, \text{Pred} \rangle$, состоящую из
 - ▶ множества **констант** Const ,
 - ▶ множества **функциональных символов** Func с сопоставленными им местностями (числами из \mathbb{N})
 - ▶ (функциональный символ \mathbf{f} местности n будет изображаться так: $\mathbf{f}^{(n)}$)
и
 - ▶ множества **предикатных символов** Pred с сопоставленными им местностями
 - ▶ (предикатный символ P местности n будет изображаться так: $P^{(n)}$)

Синтаксис **термов** логики предикатов (заданной сигнатуры) задаётся так:

$$t ::= \mathbf{c} \mid \mathbf{x} \mid \mathbf{f}^{(n)}(t_1, \dots, t_n),$$

где t, t_1, \dots, t_n — термы, $\mathbf{c} \in \text{Const}$, $\mathbf{x} \in \text{Var}$ и $\mathbf{f}^{(n)} \in \text{Func}$

Term — так будем обозначать множество всех термов

Атомарная формула (заданной сигнатуры) — это запись вида $P^{(n)}(t_1, \dots, t_n)$, где $P^{(n)} \in \text{Pred}$ и t_1, \dots, t_n — термы

Atom — так будем обозначать множество всех атомарных формул

Присваивание — это запись вида « $x := t$ », где $x \in \text{Var}$ и $t \in \text{Term}$

Тест (или, по-другому, — **условие**) — это произвольный атом

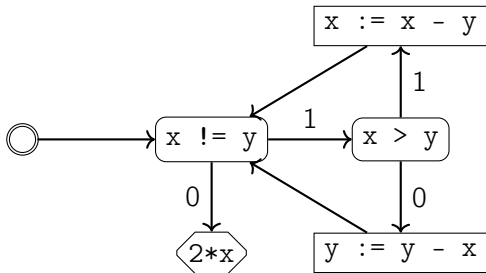
Стандартная схема программ отличается от схемы Ляпунова-Янова следующим:

1. В схеме может быть много выходов
2. Каждый выход помечен набором термов: t_1, \dots, t_n
 - ▶ (этот набор можно расценивать как результат, возвращаемый программой)
3. Каждый преобразователь помечен не операторным символом, а присваиванием
4. Каждый распознаватель помечен не логическим символом, а тестом

Пример

```
while (x != y)
  if (x > y)
    x = x - y;
  else
    y = y - x;
return 2*x
```

Этой программе соответствует такая стандартная схема:



Смысл основных элементов (примитивов) стандартной схемы задаётся интерпретацией логики предикатов, то есть системой $\mathcal{I} = \langle D, \overline{\text{Const}}, \overline{\text{Func}}, \overline{\text{Pred}} \rangle$, где:

- ▶ D — область интерпретации, или, по-другому, множество предметов
- ▶ $\overline{\text{Const}} : \text{Const} \rightarrow D$ — оценка констант
 - ▶ $\overline{c} = \overline{\text{Const}}(c) \in D$
- ▶ $\overline{\text{Func}} : \text{Func} \rightarrow \bigcup_{n=1}^{\infty} (D^n \rightarrow D)$ — оценка функциональных символов
 - ▶ $\overline{f^{(n)}} = \overline{\text{Func}}(f^{(n)}) : D^n \rightarrow D$
- ▶ $\overline{\text{Pred}} : \text{Pred} \rightarrow \bigcup_{n=1}^{\infty} (D^n \rightarrow \{0, 1\})$ — оценка предикатных символов
 - ▶ $\overline{P^{(n)}} = \overline{\text{Pred}}(P^{(n)}) : D^n \rightarrow \{0, 1\}$

Оценкой переменных (для заданной интерпретации \mathcal{I}) называется отображение вида $\xi : \text{Var} \rightarrow D$

Оценку переменных также будем называть состоянием данных схемы

$[x_1/d_1, \dots, x_n/d_n]$ — так будем обозначать состояние данных ξ , такое что $\xi(x_1) = d_1, \dots, \xi(x_n) = d_n$ и значения остальных переменных неважны

$\Xi_{\mathcal{I}}$ — так будем обозначать множество всех состояний данных для интерпретации \mathcal{I}

Значения термов t и атомарных формул A в интерпретации \mathcal{I} на оценке ξ ($t[\xi]_{\mathcal{I}}$, $A[\xi]_{\mathcal{I}}$) задаются обычным естественным индуктивным образом:

- ▶ Если $x \in \text{Var}$, то $x[\xi]_{\mathcal{I}} = \xi(x)$
- ▶ Если $c \in \text{Const}$, то $c[\xi]_{\mathcal{I}} = \bar{c}$
- ▶ Если $\mathbf{f}^{(n)} \in \text{Func}$, то $\mathbf{f}(t_1, \dots, t_n)[\xi]_{\mathcal{I}} = \bar{\mathbf{f}}(t_1[\xi]_{\mathcal{I}}, \dots, t_n[\xi]_{\mathcal{I}})$
- ▶ Если $P(t_1, \dots, t_n) \in \text{Atom}$, то $P(t_1, \dots, t_n)[\xi]_{\mathcal{I}} = \bar{P}(t_1[\xi]_{\mathcal{I}}, \dots, t_n[\xi]_{\mathcal{I}})$

Состояние вычисления стандартной схемы π в интерпретации \mathcal{I} — это пара (v, ξ) , где v — вершина схемы π и $\xi \in \Xi_{\mathcal{I}}$

Для состояния данных ξ , переменной x и предмета d записью $\xi[x \leftarrow d]$ обозначим состояние данных, получающееся из ξ присваиванием значения d в переменную x :

$$\xi[x \leftarrow d](y) = \begin{cases} d, & \text{если } y = x; \\ \xi(y) & \text{иначе} \end{cases}$$

Вычисление стандартной схемы π в интерпретации \mathcal{I} на состоянии данных ξ ($comp(\pi, \mathcal{I}, \xi)$) — это (конечная или бесконечная) последовательность состояний вычисления

$$(v_1, \xi_1), (v_2, \xi_2), (v_3, \xi_3), \dots,$$

устроенная так:

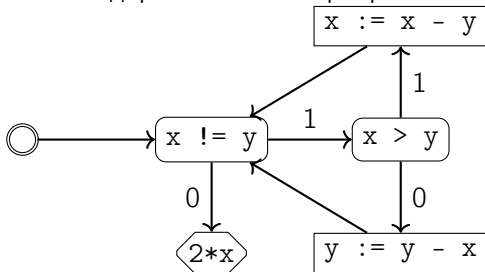
1. $v_1 = \bigcirc$, $\xi_1 = \xi$, $v_1 \rightarrow v_2$, $\xi_2 = \xi$
2. Если $v_i = \boxed{x := t}$, то $v_i \rightarrow v_{i+1}$ и $\xi_{i+1} = \xi_i[x \leftarrow t[\xi_i]_{\mathcal{I}}]$
3. Если $v_i = \boxed{P(t_1, \dots, t_n)}$, то $v_i \xrightarrow{P(t_1, \dots, t_n)[\xi_i]_{\mathcal{I}}} v_{i+1}$
4. Если $v_i = \langle t_1, \dots, t_n \rangle$, то (v_i, ξ_i) — последний элемент последовательности

Результат $val(\pi, \mathcal{I}, \xi)$ вычисления стандартной схемы π в интерпретации \mathcal{I} на состоянии данных ξ определяется так:

1. Если вычисление $comp(\pi, \mathcal{I}, \xi)$ конечно и оканчивается парой $(\langle t_1, \dots, t_n \rangle, \xi')$, то $val(\pi, \mathcal{I}, \xi) = (t_1[\xi]_{\mathcal{I}}, \dots, t_n[\xi]_{\mathcal{I}})$
2. Если вычисление $comp(\pi, \mathcal{I}, \xi)$ бесконечно, то $\bar{\pi}(\mu) = \perp$, где \perp не является набором термов

Семантика стандартной схемы π в заданной интерпретации \mathcal{I} с предметной областью D описывается отображением $\bar{\pi}_{\mathcal{I}} : \Xi_{\mathcal{I}} \rightarrow (D^* \cup \{\perp\})$, таким что $\bar{\pi}_{\mathcal{I}}(\xi) = val(\pi, \mathcal{I}, \xi)$

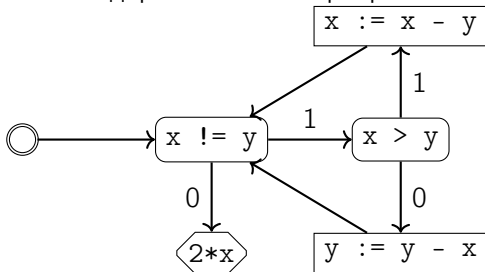
Пример вычисления стандартной схемы программ



Рассмотрим «естественную» арифметическую интерпретацию \mathcal{I}_{ar} :

- ▶ Предметы — это целые числа
- ▶ Символы $!=$, $>$, $-$, $*$ и 2 оцениваются соответственно как отношения неравенства и сравнения (строго больше) чисел, операции разности и умножения чисел и константа 2

Пример вычисления стандартной схемы программ



Вычисление этой схемы в \mathcal{I}_{ar} на оценке $[x/6, y/4]$ устроено так:

$(\odot, [x/6, y/4]), (x \neq y, [x/6, y/4]), (x > y, [x/6, y/4]),$
 $(x := x - y, [x/6, y/4]), (x \neq y, [x/2, y/4]), (x > y, [x/2, y/4]),$
 $(y := y - x, [x/2, y/4]), (x \neq y, [x/2, y/2]), (2*x, [x/2, y/2])$

Результат этого вычисления — (4)