УДК 519.6

*Э. Б. АЙНАКУЛОВ, кандидат технических наук, Ташкентский институт инженеров транспорта (Узбекистан)*

# МЕТОДИКА ОРГАНИЗАЦИИ МАШИННЫХ АЛГЕБРАИЧЕСКИХ ВЫЧИСЛЕНИЙ

Рассмотрен вопрос организации машинных вычислений, позволяющих использовать их в качестве моделей соответствующих функциональных узлов алгебраического устройства.

**Приведение сложного одночлена к нормальному виду.** Из всех видов числа $x = a^p b^q c^s = b^q a^p c^s = c^s b^q a^p = \ldots$ условно принимаем за нормальный вид числа $x$ число $x = a^p b^q c^s$, в котором основания степеней расположены в алфавитном порядке.

Пусть задан одночлен $x = b^q a^p c^s$. Требуется привести его к нормальному виду. Процедура приведения выразится следующим алгоритмом:

$$a_{\text{н}} \sim b \to a \neq b \to a_{\text{н}} \sim a \to a_{\text{н}} = a \to x :=$$
$$= a^p\, b^q\, c^s \to b_{\text{н}} \sim b \to b_{\text{н}} = b \Rightarrow x := a^p\, b^q\, c^s. \quad (1)$$

Индекс “н” у разрядных оснований $a$, $b$, $c$ указывает на то, что основания $a_{\text{н}}$, $b_{\text{н}}$, $c_{\text{н}}$ относятся к числу $x_{\text{н}}$, форма которого задана априорно.

**Приведение многочлена к нормальному виду.** Чтобы показать, как выполняется машинным путём процедура приведения многочлена к нормальному виду, обратимся к простому примеру, позволяющему увидеть общее решение. Пусть задан многочлен

$$a^{p_1} b^{q_1} c^{s_1} + a^{p_2} b^{q_2} c^{s_2}, \quad (2)$$

одночлен которого нормализован по (1). Тогда нормализацию многочлена (2) можно осуществить по алгоритму

$$a^{p_1} \sim a^{p_2} \begin{cases} \to a^{p_1} > a^{p_2} \Rightarrow (2) \\ \to a^{p_1} < a^{p_2} \Rightarrow a^{p_1} b^{q_1} c^{s_1} + a^{p_2} b^{q_2} c^{s_2} \\ \to a^{p_1} = a^{p_2} \to b^{q_1} \sim b^{q_2} \begin{cases} \to b^{q_1} > b^{q_2} \Rightarrow (2) \\ \to b^{q_1} < b^{q_2} \to 1 \\ \to b^{q_1} = b^{q_2} \to 2 \end{cases} \end{cases}$$

$$1 \to a^{p_2} b^{q_2} c^{s_2} + a^{p_1} b^{q_1} c^{s_1}$$

$$2 \to c^{s_1} \sim c^{s_2} \begin{cases} \to c^{s_1} > c^{s_2} \Rightarrow (1) \\ \to c^{s_1} < c^{s_2} \to a^{p_2} b^{q_2} c^{s_2} + a^{p_1} b^{q_1} c^{s_1} \\ \to c^{s_1} = c^{s_2} \Rightarrow (6) := 2a^{p_1} b^{q_1} c^{s_1} \end{cases} \quad (3)$$

**Сравнение сложных одночленов на подобие.** Два числа $x = a^p b^q c^s$ и $y = d^k e^l f^m$, заданные нормальными видами, являются подобными, если $a^p = d^k$, $b^q = e^l$, $c^s = f^m$.

Алгоритм определения их подобия

$$a^p \sim d^k \begin{cases} \to a^p \neq d^k \Rightarrow x \text{ и } y - \text{не подобны.} \\ \to a^p = d^k \to b^q \sim e^l \begin{cases} \to b^q \neq e^l \Rightarrow (1) \\ \to b^q = e^l \to 2 \end{cases} \end{cases}$$

$$1 \Rightarrow x \text{ и } y - \text{не подобны.} \quad (4)$$

$$2 \to c^s \sim f^m \begin{cases} \to c^s \neq f^m \Rightarrow x \text{ и } y - \text{не подобны.} \\ \to c^s = f^m \Rightarrow x \text{ и } y - \text{подобны.} \end{cases}$$

В алгоритме (4) сравнения степеней показаны в двухтактном режиме, т. е. сокращением, хотя их можно было бы разделить на операции сравнения сначала оснований степеней, а затем их показателей. Но для процедуры (3) такое разделение принципиальной роли не играет.

**Сложение подобных многочленов.** Если положить, что числа $x^p$ и $y^q$ есть одночлены вида $x = a^k b^l c^m$, $y = d^n e^p f^s$, и допустить, что они подобны в соответствии с (5), то сложение их будет определяться по формуле

$$\alpha x^p + \beta y^q = (\alpha + \beta) a^k b^l c^m. \quad (5)$$

Операция сложения многочленов сводится к операциям сложения подобных членов по (4) и неподобных членов по (6):

$$\alpha x^p + \beta y^q = \alpha a^k b^l c^m + \beta d^n e^p f^q. \quad (6)$$

**Вынесение общего множителя (буквенного) за скобки.**

Если во всех членах многочлена содержатся степени, подобные по основанию, то они делятся на наименьшую степень, выносимую затем за скобки, а в скобках остается результат от деления многочлена. Сказанное проиллюстрируем примером.

Пример 1. Пусть дан многочлен:

$$a^2bc^3 + ab^3c^2 + abc.$$

Требуется машинным путём выполнить операцию вынесения общего множителя за скобки.

*Решение.* Для удобства алгоритмизации решения всем буквенным символам присвоим индексы. Многочлен примет вид

$$a_1^2 b_1 c_1^3 + a_2 b_2^3 c_2^2 + a_3 b_3 c_3.$$

Алгоритм нахождения общих множителей в формализованном виде выразится через последовательность операций следующим образом:

$$a_1^2 \sim a_2 \to a_1^2 < a_2 \to a_2 \sim a_3 \begin{array}{l} \to a_2 = a_3 \Rightarrow \\ \to a_2 < a_3 \\ \to a_2 > a_3 \end{array}$$

$\Rightarrow a_2 \vee a_{.3}$ – общий множитель $\Rightarrow$

$$b_1 \sim b_2 \begin{array}{l} \to b_1 < b_2^2 \to b_1 \sim b_3 \begin{array}{l} \to b_1 = b_3 \Rightarrow \\ \to b_1 > b_2^2 \\ \to b_1 < b_3 \end{array} \\ \to b_1 \neq b_2 \\ \to b_1 < b_2 \end{array}$$

$\Rightarrow b_2 \vee b_{.3}$ – общий множитель $\Rightarrow$

$$c_1^3 \sim c_2^2 \begin{array}{l} \to c_1^3 < c_2^2 \to c_2^2 \sim c_3 \begin{array}{l} \to c_2^2 > c_3 \Rightarrow \\ \to c_2^2 \neq c_3 \\ \to c_2^2 < c_3 \end{array} \\ \to c_1^3 = c_2^2 \\ \to c_1^3 < c_2^2 \end{array}$$

$c_3$ – общий множитель $\Rightarrow OM :=$

$= (a_2 \vee a_3)\cdot(b_1 \vee b_3)\cdot c_3 \Rightarrow (1)/abc = ? \Rightarrow (1) :=$
$abc(ac^2 + b^2c + 1).$

**Разложение квадратного трёхчлена на множители.** Эта задача давно решена на современных ЭВМ программными средствами. Здесь предлагается программное решение рассматривать как аппаратное за счет использования для вшитых программ, или встроенных устройств, а решение задачи рассматривать как самостоятельную машинную операцию. Процедуру разложения квадратного трехчлена на множители покажем на следующем примере.

Пусть задан квадратный трёхчлен

$$Э = ax^2 + bx + c.$$

Тогда алгоритм разложения его на множители в множестве действительных чисел представлен следующим образом:

$$b^2 = ? \to 4a =? \to 4ac =? \to b^2 - 4ac =? \to$$

$$b^2 - 4ac := D \to D \sim 0 \to D > 0 \begin{array}{l} \to \sqrt{D} = ? \to 1 \\ \to D = 0 \to 2 \\ \to D < 0 \Rightarrow x_1 = x_2 = \varnothing. \end{array}$$

$1 \to 2a =? \to -b + \sqrt{D} = ? \to (-b + \sqrt{D}) : 2a =$

$= ? \to -b - \sqrt{D} = ? \to (-b - \sqrt{D}) : 2a = ? \Rightarrow$

$x_1 := (-b + \sqrt{D})/2a \Rightarrow x_2 := (-b - \sqrt{D})/2a \Rightarrow$

$Э := (x - x_1)\cdot(x - x_2).$

$$2 \to 2\cdot a = ? \Rightarrow x_1 := x_2 := -b/2a \Rightarrow Э := (x - x_1)^2. \quad (7)$$

Определение корней квадратного трехчлена в Э примера 2 фактически осуществлено операциями в алгоритме (7):

$$x_1 := (-b + \sqrt{D})/2a, \; x_2 := (-b - \sqrt{D})/2a.$$

В дальнейшем операцию определения корней квадратного уравнения (трехчлена) будем рассматривать как самостоятельную и включим в набор машинных команд.

**Решение уравнений первой степени.** Пусть имеется линейное уравнение 1-й степени $ax + b = cx + d$, в котором требуется определить $x$. Тогда машинное решение через операции машинной алгебры представляется следующим образом:

$$(ax + b = cx + d) -^{p} cx \to (ax - cx + b = d) -^{p} b \to$$
$$\to (ax - cx = d - b) \to (ax - cx = d - b) \to$$
$$\to \langle ВМ \rangle\, (ax - cx = d - b) \to (x\,(a - c) =$$
$$= (d - b)) :^{p} (a - c) \Rightarrow x := (d - b) : (a - c). \quad (8)$$

В алгоритме (8) опущены операции сравнения оснований степеней на существование, без которых не проводится решение уравнения. Они были опущены по причине лимитного объёма данной работы. Кроме этого, в (8) встречается символ <ВМ>, обозначающий операцию занесения общего множителя за скобки.

**Организация команд второго порядка машинной алгебры.** Рассмотренные выше сложные алгебраические операции (вычисления) можно принять в набор команд второго порядка, являющиеся производными от основных, которые представлены в [1, 2].

Команды (операции) второго порядка предназначены для выполнения более сложных машинных вычислений. Они бывают унарными и бинарными. Унарные операции будем обозначить соответствующими символами.

В таблице 1 приведены операции или команды, которые описывались выше, а также и другие, выполнимость которых актуальна.

*Таблица 1* – **Операции (команды) 2-го порядка машинной алгебры**

| Наименование операции | Символ операции | Форма команды | Примечание |
|---|---|---|---|
| Нормализация одночлена, нормализация многочлена | < *НО* > | < *НО* >*x* | *x* – одночлен |
| Сравнение одночленов на подобие | < *НМ* > | < *НМ* >*P*(*a*,*b*,*c*) | *P* – многочлен, *a*,*b*,*c* – переменные степени и т. п. |
| Сложение подобных членов, вычитание подобных членов | $\sim^n$ | $x \sim^n y$ | *x*, *y* – сложные одночлены |
| Умножение подобных членов | $+^n$ | $x +^n y$ | *n* – подобие |
| Деление подобных членов, сложение многочленов | $-^n$ | $x -^n y$ | *x*, *y* – одночлены |
| Вычитание многочленов, умножение многочленов | $*^n$, $(x^n)$ | $x *^n y$ | |
| Деление многочленов | $:^n$ | $x :$ | |
| Вынесение общего множителя за скобки | $(/^n)$ | $^n y$ | |
| | $+^m$ | $P +^m Q$ | *P*, *Q* – многочлены |
| | $-^m$ | $P -^m Q$ | |
| Разложение квадратного трёхчлена на множители | $x^m$, $(*^m)$ | $P *^m Q$ | |
| | $:^m$, $(/^m)$ | $P :^m Q$ | |
| Определение корней квадратного уравнения | < *ВМ* > | < *ВМ* >*P* | *P* – многочлен |
| Определение корня линейного уравнения | < *Р2* > | < *Р2* > *F*(*x*) | *F(x)* – квадратный трёхчлен $ax^2 + bx + c$ |
| Возведение одночлена в степень | < *К2* > | < *К2* > *F*(*x*) | |
| Извлечение квадратного корня из одночлена | < *К1* > | < *К1* > *F*(*x*) | *F*(*x*) – линейная функция |
| | ** | $x ** n$ | *x* – одночлен, *n* – показатель степени |
| | √ | $\sqrt{x}$ | *x* – одночлен, $x \geq 0$ |

**Список литературы**


1 **Aynakulova, T. S.** Synthesis of Algebraic Devices / T. S. Aynakulova, B. S. Utabaev, E. B. Aynakulov // International Conference on IT Promotion in Asia 2007 in conjunction with International Summit on Information and Communication Technologies. September 24 – 28, 2007. – Tashkent : University of IT Tashkent, Uzbekistan, 2007. – P. 236–242.

2 **Aynakulov, E. B.** Principles of Analysis of Algebraic Structures / E. B. Aynakulov // Fifth International Conference on Application of Fuzzy Systems and Soft Computing. ICAFS – 2002. September 17–18, 2002. – Milan, Italy, 2002. – P. 203–206.



Получено 20.09.2009


**E. D. Aynakulov.** Technique of machine algebraic calculations.


We consider the organization of machine computations to make use of them as models of the functional units of algebraic systems.