

## 10. Программирование в пакете MathCAD

Существуют два способа программирования в MathCAD:

- программирование в пакете MathCAD без использования программных модулей;
- программирование с использованием программных модулей.

**Первый способ** (в дальнейшем для простоты названный безмодельным программированием) реализуется записью соответствующих конструкций непосредственно в математических областях документа MathCAD, и он приемлем для сравнительно простых алгоритмов.

**Второй способ** (называемый для простоты модульным программированием) предполагает реализацию отдельных независимых алгоритмов вычисления (например, решение нелинейного уравнения методом «деления отрезка пополам») в виде отдельных программных модулей, которые будем называть программами-функциями (сокращенно П-Ф).

### БЕЗМОДУЛЬНОЕ ПРОГРАММИРОВАНИЕ В ПАКЕТЕ MathCAD

#### Программирование линейных алгоритмов

Характерной особенностью линейных алгоритмов является строго последовательное выполнение всех операций алгоритма без пропусков и повторений вычислений. Поэтому конструкции, реализующие такой алгоритм, записываются в документе MathCAD в нужном порядке их выполнения, т.е. «слева-направо – сверху-вниз».

**Пример 1.** Составить программу для вычисления корней квадратного уравнения:  $ax^2 + bx + c = 0$  по известной формуле:

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (1)$$

Алгоритм (1) является линейным и фрагмент документа MathCAD содержит конструкции, приведенные на рис 1.

$$\begin{aligned} a &:= 2 & b &:= 5 & c &:= 8 & d &:= \sqrt{b^2 - 4 \cdot a \cdot c} \\ x_1 &:= \frac{-b + d}{2 \cdot a} & x_2 &:= \frac{-b - d}{2 \cdot a} \\ x_1 &= -1.25 + 1.561i & x_2 &= -1.25 - 1.561i \end{aligned}$$

Проверка найденных корней

$$a \cdot x_1^2 + b \cdot x_1 + c = 0.00000 \quad a \cdot x_2^2 + b \cdot x_2 + c = 0.00000$$

Программирование разветвляющихся алгоритмов

Характерной чертой разветвляющихся алгоритмов является наличие в них нескольких возможных ветвей вычислений. Выбор конкретной ветви зависит от выполнения (или не выполнения) заданных условий на значения переменных алгоритма.

**Пример 2.** Значение переменной  $y$  зависит от значений переменной  $x$  и определяется выражением:

$$y = \begin{cases} x^2, & \text{если } x \leq 0; \\ \sqrt{x}, & \text{в противном случае.} \end{cases}$$

Для реализации разветвляющегося алгоритма необходимо использовать:

- конструкции, проверяющие выполнение заданных условий;
- конструкции, выбирающие нужную ветвь вычислений в зависимости от результатов проверки заданных условий.

Для проверки заданных условий в MathCAD используется: *выражение отношений, логические операции и логические выражения.*

**Выражением отношений** (или просто *отношением*) называется конструкция вида:

$$\langle \text{выр.1} \rangle \langle \text{операция отношения} \rangle \langle \text{выр.2} \rangle,$$

где  $\langle \text{выр.1} \rangle$ ,  $\langle \text{выр.2} \rangle$  - произвольные арифметические выражения,  $\langle \text{операция отношения} \rangle$

- любая из следующих операций:  $\langle | \langle | \leq | \rangle | \geq | \neq | = | \rangle$  (здесь вертикальные чёрточки являются разделительным символом при перечислении).

Для ввода знаков операций отношений можно использовать палитру БУЛЕВО

(приведённую на рис.3) или использовать клавиши, обозначения которых приведены в таб. 1.



Палитра инструментов БУЛЕВО

**Внимание!** Не следует путать знак операции сравнения = с похожим знаком вывода значений переменных. Знак операции = имеет больший размер и более жирное начертание.

Таблица 1

Знаки операции	Клавиши

<	[<]
≤	[Ctrl] + [9]
>	[>]
≥	[Ctrl] + [0]
=	[Ctrl] + [=]
≠	[Ctrl] + [3]

Выражение отношений принимает одно из двух значений: 1 - если заданное отношение выполняется или 0 – в противном случае. Значение 1 можно интегрировать как значение ИСТИНА, а 0 – как ЛОЖЬ.

**Задание 1.** Пусть значения целой переменной  $x = 3$ . Определить значение следующих выражений отношений:

**а)  $x \geq 4$ ;    б)  $x + 1 \geq 4$  ;    в)  $x - 4 > 1$**

Для проверки более сложных условий используются *четыре логических операций*, обозначение которых приведены в табл. 2

**Таблица 2**

Название операции	Знак
Логическое отрицание (NOT)	$\neg$
Логическое ИЛИ (OR)	$\vee$
Логическое И (AND)	$\wedge$
Исключающее ИЛИ (XOR)	$\oplus$

Знаки этих операций вводятся с палитры БУЛЕВО. Результат выполнения этих операций, приведен в табл. 3.

**Таблица 3**

NOT $\neg$	AND $\wedge$	OR $\vee$	XOR $\oplus$
------------	--------------	-----------	--------------

$0 \neg = 1$	$0 \wedge 0 = 0$	$0 \vee 0 = 0$	$0 \oplus 0 = 0$
$1 \neg = 0$	$0 \wedge 1 = 0$	$0 \vee 1 = 1$	$1 \oplus 0 = 1$
	$1 \wedge 0 = 0$	$1 \vee 0 = 1$	$0 \oplus 1 = 1$
	$1 \wedge 1 = 1$	$1 \vee 1 = 1$	$0 \oplus 0 = 0$

*Логическим выражением* называется конструкция, состоящая из выражений отношений, логических операций и круглых скобок. Логическое выражение принимает только одно из двух значений: 1 или 0; вычисляется слева направо с учетом приоритета входящих в выражение операции. Наивысший приоритет круглые скобки, а затем по убыванию: AND, OR и XOR – одинаковый приоритет и самый низкий приоритет выражения отношений.

**Задание 2.** Определите порядок вычисления значений логических выражений в документе MathCAD, приведенных на рисунке 3

$$x := 0.2 \quad y := -4$$

$$(x > 3) \wedge (y, 0) = 0$$

$$\neg(x > 3) \wedge (y < 0) = 1$$

**Рис. 3** Примеры логических выражений

**Пример 2.** Записать логическое выражение, принимающее значение 1 при попадании точки с координатами  $(x, y)$  в первую четверть.

Логическое выражение имеет вид

$$(x > 0) \wedge (y > 0)$$

Для выбора нужной ветви разветвляющегося алгоритма используется конструкция, названная *условной функцией if*, записывается в виде:

$$if (<логическое выражение>, <выр. 1>, <выр. 2>),$$

где имя функции *if* вводится с клавиатуры.

Если логическое выражение равно 1, то значение функции определяется *выр. 1*, в противном случае – *выр. 2*. Блок-схема этой функции приведена на рис. 4.

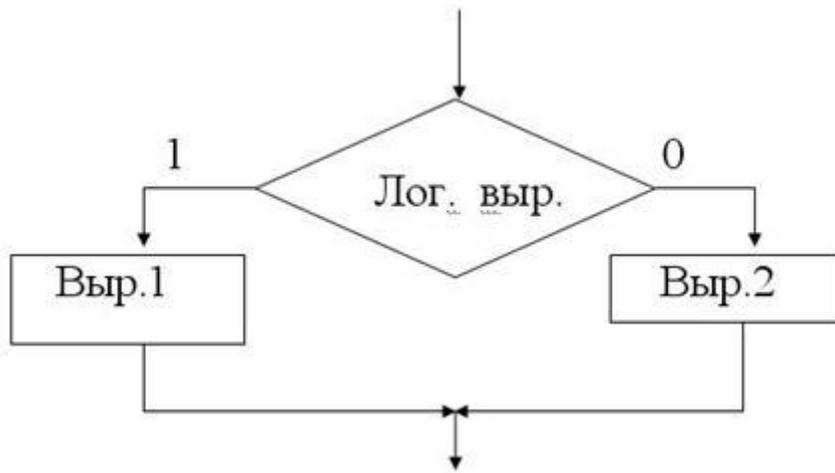


Рис. 4. Блок-схема функции *if*

При программировании разветвляющихся алгоритмов с тремя и более вычислительных ветвей на месте *выр. 1* и *выр. 2* вновь может использоваться функция *if*.

**Пример 3.** Используя условную функцию *if* запрограммировать два разветвляющихся алгоритма:

а) 
$$y(x) = \begin{cases} x^2, & \text{если } x \leq 0; \\ \sqrt{x}, & \text{если } x > 0. \end{cases}$$

б) 
$$z(x) = \begin{cases} 30, & \text{если } x \leq -1; \\ |x|, & \text{если } -1 < x \leq 1; \\ x^2 - 30, & \text{если } x > 1. \end{cases}$$

**$x := 3$**

**$y := if(x \leq 0, x^2, \sqrt{x}) \quad y = 1.732$**

**$z(x) := if(x \leq -1, 30, if(-1 \leq x \leq 1, |x|, x^2 - 30))$**

**$x := -10, -9.95.. 10$**

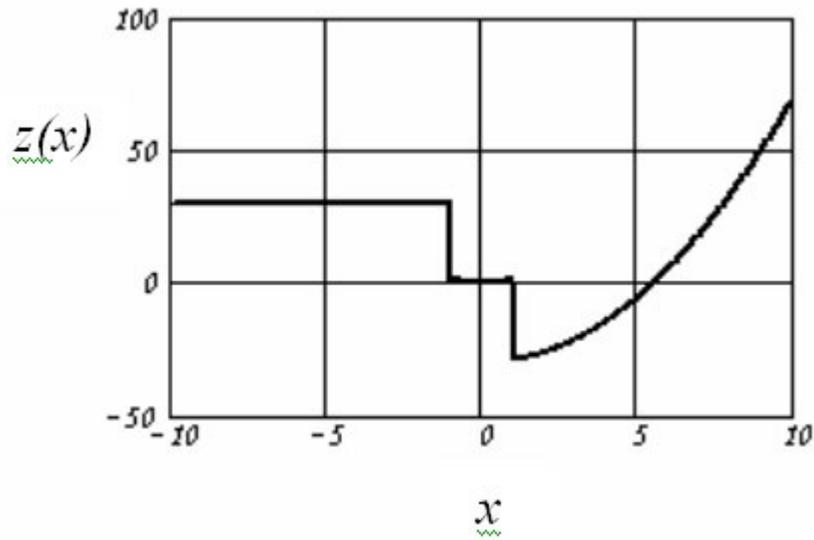


Рис. 5. Реализация разветвляющихся алгоритмов

В MathCAD имеется ряд встроенных функций, которые возвращают результат, зависящий от знака или величины аргумента и которые могут использоваться при программировании разветвляющихся алгоритмов. Приведем некоторые из них:

- $\mathit{ceil}(x)$  – наименьшее целое, большее или равное  $x$ ;
- $\mathit{trunc}(x)$  – целая часть вещественного числа  $x$ ;
- $\mathit{floor}(x)$  – наибольшее целое, меньшее или равное  $x$ ;
- $\mathit{round}(x, n)$  – округленное значение вещественного  $x$  с точностью до  $n$  знаков после десятичной точки;
- $\Phi(x)$  – функция Хевисайда – равна 0 при  $x < 0$  и 1 в противном случае;
- $\mathit{sign}(x)$  – функция знака (равна 0 если  $x = 0$ ; -1, если  $x < 0$  и 1, если  $x > 0$ );
- $\mathit{signum}(x)$  – возвращает 1, если  $x = 0$  и  $\frac{x}{|x|}$  в остальных случаях.

### Программирование циклических алгоритмов

**Типы циклов.** По способам организации цикла можно выделить:

- а) *цикл типа арифметической прогрессии;*
- б) *итерационный цикл.*

#### Программирование цикла типа арифметической прогрессии

Особенностью цикла типа арифметической прогрессии является изменение параметра цикла по закону арифметической прогрессии и поэтому можно, не выполняя цикла, определить количество повторений цикла.

Параметр такого цикла задается дискретной переменной и тогда конструкции, входящие в тело цикла располагаются, начиная от этого описания и до конца документа MathCAD или до конструкции, переопределяющей дискретную переменную – параметр цикла.

		$n := 10$		$i := 1..10$							
		$z_i := \frac{1}{i+4}$									
$z^T =$		1	2	3	4	5	6	7	8	9	10
	1	0.2	0.167	0.143	0.125	0.111	0.1	0.091	0.083	0.077	0.071

$$B_{i,j} = \frac{1}{i+j+1}, \quad i = 1, \dots, n, \quad j = 1, \dots, m$$

Этот цикл имеет уже два параметра:  $i$  – параметр, определяющий номер строки;  $j$  – параметр, определяющий номер столбца матрицы. Заметим, что такой цикл называется *двойным* циклом.

$n := 3$		$m := 6$	
$i := 1..n$		$j := 1..m$	
$B_{i,j} := \frac{1}{i+j+1}$		$B =$	
		$\begin{pmatrix} 0.333 & 0.25 & 0.2 & 0.167 & 0.143 & 0.125 \\ 0.25 & 0.2 & 0.167 & 0.143 & 0.125 & 0.111 \\ 0.2 & 0.167 & 0.143 & 0.125 & 0.111 & 0.1 \end{pmatrix}$	

Заметим, что если параметр цикла не входит в индексные выражения элементов массива, то он может принимать вещественные значения.

Заметим, что во фрагментах, приведенных на рисунках системная переменная  $ORIGIN=1$ .