



ПРОГРАММИРОВАНИЕ В ПАКЕТЕ MATHCAD

Программирование в пакете MATHCAD

В пакете MathCad могут быть реализованы два способа программирования:

- *безмодульное программирование* – программирование без использования программных модулей;
- *модульное программирование* – программирование с использованием программных модулей.

Безмодульное программирование реализуется записью соответствующих конструкций непосредственно в математических областях документа MathCAD, и он приемлем для сравнительно простых алгоритмов.

Модульное программирование предполагает разработку отдельных независимых алгоритмов в виде программных модулей, которые реализуются в виде

Безмодульное программирование

Рассмотрим реализацию трех конструкций алгоритмов: *линейную, разветвляющуюся и циклическую.*

Программирование линейных алгоритмов

Конструкции, реализующие линейный алгоритм, записываются в документе MathCAD последовательно строго в порядке их выполнения, т. е. «**слева-направо**» и «**сверху-вниз**».

Пример. Составить программу вычисления площади треугольника по формуле:

$$S = \sqrt{p(p - a)(p - b)(p - c)} ,$$

где p – полупериметр; a, b, c – стороны треугольника.

Исходные данные: $a = 1.6$; $b = 2.03$; $c = 0.5$

Программирование линейных алгоритмов

Решение

$$a := 1.6 \quad b := 2.03 \quad \underline{\underline{c}} := 0.5$$

$$p := \frac{a + b + c}{2}$$

$$\underline{\underline{S}} := \sqrt{p \cdot (p - a) \cdot (p - b) \cdot (p - c)}$$

$$S = 0.229$$

Программирование разветвляющихся алгоритмов

Характерной чертой разветвляющихся алгоритмов является наличие в них нескольких ветвей вычислений. Выбор конкретной ветви зависит от выполнения заданного условия.

Условие реализуется с помощью *логического выражения*.

Логическое выражение может включать в себя отношения, логические константы и переменные, логические функции, логические операции и круглые скобки.

Для ввода знаков операций отношений и логических операций используется **Панель логики**.

Результатом вычисления логического выражения является либо **1 (ИСТИНА)**, либо **0 (ЛОЖЬ)**

Отношения

Отношение – это простейшее логическое выражение.

Операции отношений: $> | \geq | < | \leq | = | \neq$

Необходимо помнить! Операции отношений имеют самый низкий приоритет, поэтому в сложных логических выражениях отношение всегда заключается в *круглые скобки*.

Замечание. Знак сравнения (=) нельзя путать со знаком численного вывода (=). Знак сравнения имеет более жирное начертание и вводится с логической панели инструментов.

$$x := y \quad y := -2$$

$$z := 6$$

Примеры вычисления отношений

$$x > y = 1$$

$$x + 3 = z = 1$$

$$0 \leq x \leq 5 = 1$$

$$x - 2 \cdot y \geq 3 \cdot z = 0$$

Сложные логические выражения

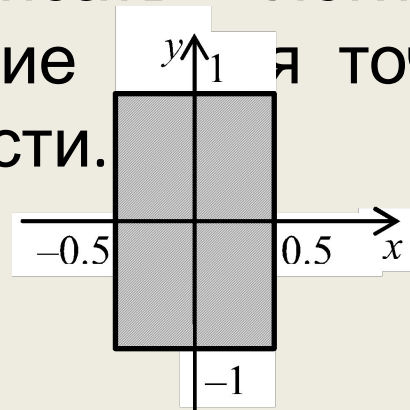
Логические операции : \neg (NOT) | \wedge (AND) | \vee (OR) | \oplus (XOR)

Примеры логических выражений

Пример 1. Указать порядок действий при вычислении логического выражения: $\neg (a \vee b) \wedge c$: $\boxed{2}$ $\boxed{6}$ $\boxed{5}$ $\boxed{4}$ $\boxed{3}$ $\boxed{1}$

$$(i \neq 1) \vee a \wedge \neg (b \wedge (j > k))$$

Пример 2. Записать логическое выражение, принимающее значение 1 для точек, принадлежащих заштрихованной области.



$$(-0.5 \leq x \leq 0.5) \wedge (-1 \leq y \leq 1)$$

Условная функция *if*

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

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

Имя функции *if* вводится с клавиатуры.

Условная функция *if* реализует структуру «ЕСЛИ-ТО-ИНАЧЕ». Если логическое выражение равно **1**, то значение функции определяется **выр.1**, в противном случае – **выр. 2**.

При программировании разветвляющихся алгоритмов с тремя и более вычислительными ветвями в качестве **выр. 1** и **выр. 2** вновь можно использовать условную функцию *if*. Такая конструкция образует вложенную структуру.

Примеры программирования разветвляющихся алгоритмов

Пример 1. Вычислить значение y по одной из двух ветвей.

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

Решение

$$x := -7 \quad y := \text{if}(x \leq 0, x^2, \sqrt{x}) \quad y = 49$$

$$\underline{x} := 144 \quad \underline{y} := \text{if}(x \leq 0, x^2, \sqrt{x}) \quad y = 12$$

Примеры программирования разветвляющихся алгоритмов

Пример 2. Вычислить значение z по одной из
трех ветвей:

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

Решение

При решении используется вложенная
конструкция.

$$x := -3 \quad z := \text{if}(x \leq -1, 30, \text{if}(x \leq 1, |x|, x^2 - 30)) \quad z = 30$$

$$\overset{\text{w}}{x} := -0.5 \quad \overset{\text{w}}{z} := \text{if}(x \leq -1, 30, \text{if}(x \leq 1, |x|, x^2 - 30)) \quad z = 0.5$$

$$\overset{\text{w}}{x} := 3 \quad \overset{\text{w}}{z} := \text{if}(x \leq -1, 30, \text{if}(x \leq 1, |x|, x^2 - 30)) \quad z = -21$$

Примеры программирования разветвляющихся алгоритмов

Пример 3. Вычислить значение $y = \max(a, b, c)$.

Решение

Рассмотрим использование условной функции для реализации структуры «ЕСЛИ-ТО».

```
a := 2      b := 34      c := 7
y := a
y := if (b > y , b , y)
y := if (c > y , c , y)
y = 34
```

Использование функций

Встроенные функции

В Mathcad имеется большой набор встроенных функций, которые можно использовать при программировании, например:

$\mathit{trunc}(x)$ – выделение целой части числа;

$\mathit{round}(x)$ – округление вещественного числа x с точностью до n знаков после запятой;

$\Phi(x)$ – функция Хэвисайда, значение которой равно 1 при $x > 0$ и 0 в противном случае.

Функция пользователя

Функция пользователя имеет следующее описание :

$\langle \text{Имя_функции} \rangle (\langle \text{Список_формальных_аргументов} \rangle) :=$
 $\langle \text{Выражение} \rangle$

Примеры программирования функции пользователя

Пример 1. Составить описание функции, вычисляющей расстояние между двумя точками A и B на плоскости.

$$\text{Rast}(x_A, y_A, x_B, y_B) := \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$$

$$d := \text{Rast}(2, 5, 0.4, 5) = 1.6$$

Пример 2. Ввести функцию $y(x) = \begin{cases} e^x, & \text{если } x \leq 0; \\ \ln x, & \text{если } x > 0. \end{cases}$ и вычислить значения функции $y(x)$:

$$y(x) := \text{if}(x \leq 0, e^x, \ln(x))$$

$$y(-3) = 0.05 \quad y(2.71) = 0.997$$

Программирование циклических алгоритмов

По способам организации цикла можно выделить:

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

Программирование цикла типа арифметической прогрессии

Особенностью данного типа цикла является изменение *параметра цикла* по закону арифметической прогрессии.

Параметр цикла типа арифметической прогрессии задается *дискретной переменной*.

Для такого цикла заранее можно определить

количество повторений цикла

Примеры программирования циклических алгоритмов

Пример 1. Сформировать вектор z из n элементов, определяемых по правилу:

$$z_i = \frac{1}{i + 4}$$

Решение

ORIGIN := 1

n := 5

i := 1..n

$z_i := \frac{1}{i + 4}$

$z^T = (0.2 \quad 0.167 \quad 0.143 \quad 0.125 \quad 0.111)$

Примеры программирования циклических алгоритмов

Пример 2. Вычислить значение функции:

$$y(x) = \frac{\ln|x|}{a^2 + b^2}$$

для всех значений x , изменяющихся от **0.5** до **2.5** с шагом **0.2**; переменные a , b – заданные вещественные числа. *Решение*

$a := 2.1$

$b := 4.56$

$x := 0.5, 0.7 .. 2.5$

$$y(x) := \frac{\ln(|x|)}{a^2 + b^2}$$

$y(x) =$

-0.028
-0.014
$-4.18 \cdot 10^{-3}$
$3.782 \cdot 10^{-3}$
0.01
0.016
0.021
0.025
0.029
0.033
0.036

Примеры программирования циклических алгоритмов

Пример 3. Для переменной x , изменяющейся от 1 до 2 с шагом 0.2, сформировать вектор q , состоящий из соответствующих значений функции:

$$y(x) = x^2 + 1$$

Решение

ORIGIN := 1

$$n := \text{trunc}\left(\frac{2 - 1}{0.2}\right) + 1 = 6$$

$$j := 1..n \quad x_j := 1 + (j - 1) \cdot 0.2 \quad q_j := (x_j)^2 + 1$$

$$q^T = (2 \quad 2.44 \quad 2.96 \quad 3.56 \quad 4.24 \quad 5)$$

Программирование итерационных циклов

В итерационных циклах переменная, управляющая циклом, изменяется по более сложному закону, поэтому для итерационных циклов невозможно предсказать число повторений цикла (итераций).

С помощью итерационных циклов в вычислительной математике ищется приближенное решение задачи с заданной степенью точности. Точность определяется малым положительным числом ε (порядка 10^{-8} \square 10^{-6}).

Для программирования итерационных циклов в пакете Mathcad используется функция, которая записывается в виде:

until(выр.1, выр.2)

Эта функция принимает значение ***выр.2***, если ***выр.1*** ≥ 0 . В случае, когда ***выр.1*** < 0 , функция приобретает значение 0 и выполнение функции прекращается.

В качестве приближенного решения принимается значение функции, полученное на предпоследней

Программирование итерационного цикла

Пример

Вычислить \sqrt{a} приближенное значение корня квадратного по итерационной формуле:
$$x_{n+1} = 0.5 \cdot \left(x_n + \frac{a}{x_n} \right), n = 0, 1, \dots; x_0 = a.$$

x_{n+1} в качестве приближенного значения принимается, удовлетворяющее условию:
$$\left| x_{n+1}^2 - a \right| \leq \varepsilon$$

где ε – заданная точность вычисления корня квадратного.

Программирование итерационного цикла

$a := 9$

Решение

$x_0 := a$

$\varepsilon_{\text{MW}} := 10^{-6}$

$i := 0..10$

$x_{i+1} := \text{until} \left[\left| (x_i)^2 - a \right| - \varepsilon, \frac{x_i + \frac{a}{x_i}}{2} \right]$

$x^T = (9 \ 5 \ 3.4 \ 3.024 \ 3 \ 3)$