



ПРОГРАММИРОВАНИЕ В ПАКЕТЕ MATHCAD

Программирование в пакете MATHCAD

В пакете MathCad могут быть реализованы два способа программирования:

- ❖ **Безмодульное программирование** — реализуется записью соответствующих конструкций непосредственно в математических областях документа MathCAD, и он приемлем для сравнительно простых алгоритмов.
- ❖ **Модульное программирование** — предполагает разработку отдельных программных **модулей**, которые реализуются в виде **подпрограмм-функций** (сокращенно **П-Ф**).

Безмодульное программирование

Безмодульное программирование

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

Решение

$$a := 1.6 \quad b := 2.03 \quad c := 0.5$$

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

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

$$S = 0.229$$

Безмодульное программирование

Программирование разветвляющихся алгоритмов

Условная функция *if*

Для выбора нужной ветви разветвляющегося алгоритма используется конструкция, названная условной функцией ***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***.

Безмодульное программирование

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

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

$$y(-7) = 49 \quad y(144) = 12$$

Безмодульное программирование

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.

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

$$z(-2) = 30 \quad z(-0.5) = 0.5 \quad z(2) = -26$$

Безмодульное программирование

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)

$y = 34$

Безмодульное программирование

Программирование циклических алгоритмов

Программирование цикла типа арифметической прогрессии

Используется структура цикла с параметром.

Параметр цикла типа арифметической прогрессии задается *дискретной переменной*.

Для такого цикла заранее можно определить количество повторений цикла.

Безмодульное программирование

Пример алгоритма табулирования функции.

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**

$a := 2.1$

$b := 4.56$

$x := 0.5, 0.7 .. 2.5$

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

$y(x) =$

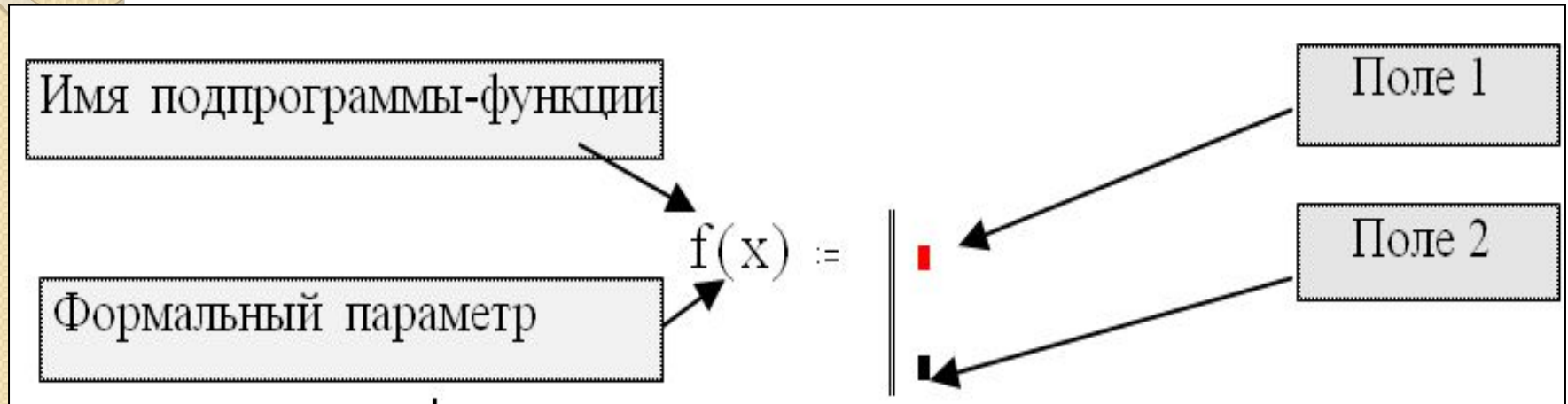
-0.028
-0.014
$-4.18 \cdot 10^{-3}$
$3.782 \cdot 10^{-3}$
0.01
0.016
0.021
0.025
0.029
0.033
0.036

Модульное программирование

Модульное программирование

В пакете MathCad программный модуль реализуется с помощью подпрограммы-функции (П-Ф).

Структура П-Ф



*Вертикальная черта и вертикальный столбец с полями для ввода операторов, образующих тело П-Ф, появляются при щелчке на кнопке **Add Line**, расположенной на **Панели программирования**.*

***Тело** П-Ф включает в себя любое число операторов (локальных операторов присваивания, условных операторов и операторов цикла), а также вызов других П-Ф и функций пользователя.*

Модульное программирование

Замечание 1. П-Ф может не иметь формальных параметров, и тогда данные передаются через имена переменных, заданных выше описания П-Ф.

Замечание 2. Самое нижнее поле ввода в структуре П-Ф служит для записи переменной или выражения, определяющих возвращаемое через имя П-Ф результат.

Замечание 3. Если результатом работы П-Ф являются несколько величин, то из них в теле П-Ф необходимо сформировать массив и его имя поместить в нижнее поле П-Ф.

Модульное программирование

Обращение к подпрограмме-функции

Вызов П-Ф располагается ниже или левее описания П-Ф и имеет вид:

< имя П-Ф > (< список фактических параметров >)

Фактические параметры определяют конкретные значения формальных параметров, при которых выполняются вычисления в П-Ф.

К моменту вызова П-Ф фактические параметры *должны быть определены*.

Локальный оператор присваивания

Для задания внутри П-Ф значения или выражения какой-либо переменной используется **локальный оператор присваивания**, имеющий вид:

< имя переменной > ← < выражение >

Программирование линейных алгоритмов

Пример. Составить П-Ф для вычисления значения функции $z(x)$, определяемую выражением:

$$z(x) = ay^{5+x} + b \cos|y|, \text{ где } y = b \sin^2 x + b^2 \sin^4 x^2$$

Вычислить значения функции при:

1) $a = 1.2;$ $b = 3;$ $x = 0.45;$

2) $a = 1.2;$ $b = 3;$ $x = -8.34$

Решение

$$a := 1.2$$

$$b := 3$$

$$z(x) := \begin{cases} y \leftarrow b \cdot \sin(x)^2 + b^2 \cdot \sin(x^2)^4 \\ a \cdot y^{5+x} + b \cdot \cos(|y|) \end{cases}$$

$$z(0.45) = 2.569$$

$$z(-8.34) = -2.588$$

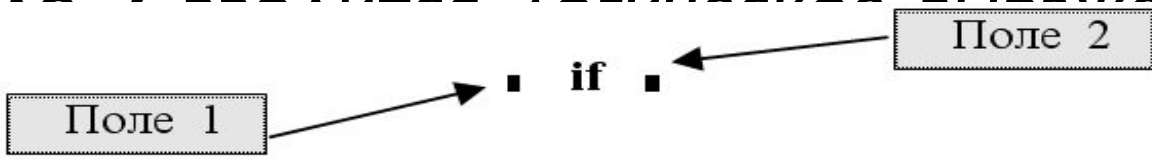
Программирование разветвляющихся алгоритмов

Для программирования разветвляющихся алгоритмов в П-Ф используется **условный оператор if**

Реализация структуры ЕСЛИ-ТО

Для ввода условного оператора надо **щелкнуть на кнопке if** на **Панели программирования**.

В **Поле 1** вводится логическое выражение (условие).

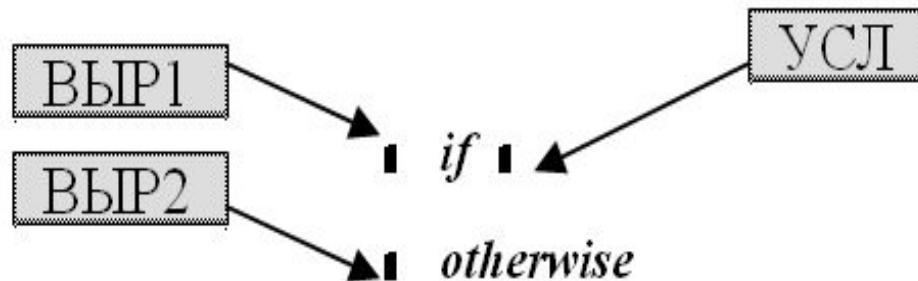


if

В **Поле 1** вводится **выражение** или **локальный оператор присваивания**, выполняемые только в том случае, когда **логическое выражение равно 1**

Реализация структуры ЕСЛИ-ТО-ИНАЧЕ

Используются условный оператор **if** и оператор **otherwise**.



Для ввода операторов надо щелкнуть на кнопках **if** и **otherwise** на **Панели программирования**.

Конструкция **ВЫР1**, стоящая перед оператором **if**, выполняется, если логическое выражение (условие) равно **1 (ИСТИНА)**.

Конструкция **ВЫР2**, стоящая перед оператором **otherwise**, выполняется, если логическое выражение (условие) равно **0 (ЛОЖЬ)**.

Примеры программирования разветвляющихся алгоритмов

Пример 1. Составить описание П-Ф для вычисления функции $\mu(x, \varepsilon)$ по формуле:

$$\mu(x, \varepsilon) = \begin{cases} \frac{1}{2\sqrt{x+1}}, & \text{если } |x - y| < \varepsilon; \\ \frac{1}{3}\sqrt[3]{x+y}, & \text{иначе где } y = |x| \end{cases}$$

Решение

Использование структуры ЕСЛИ-ТО-ИНАЧЕ

$$\mu(x, \varepsilon) := \begin{cases} y \leftarrow |x| \\ \frac{1}{2\sqrt{x+1}} & \text{if } |x - y| < \varepsilon \\ \frac{\sqrt[3]{x+y}}{3} & \text{otherwise} \end{cases}$$

$$\mu(12.8, 4) = 0.135 \quad \mu(-12.8, 4) = 0$$

Примеры программирования разветвляющихся алгоритмов

Пример 2. Составить описание П-Ф для вычисления значения 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

$$z(x) := \begin{cases} 30 & \text{if } x \leq -1 \\ |x| & \text{if } -1 < x \leq 1 \\ x^2 - 30 & \text{otherwise} \end{cases}$$

$$z(-3) = 30 \quad z(-0.5) = 0.5 \quad z(3) = -21$$

Выражение, в операторе **otherwise** будет вычисляться только в том случае, когда не выполняются условия **в двух вышестоящих операторах if**.

Примеры программирования разветвляющихся алгоритмов

Пример 3. Даны два числа x , y . Составить описание П-Ф, которая переменной x присваивает максимальное значение из этих двух чисел, а y – минимальное.

$P(x, y) :=$

	if $x > y$		$v_0 \leftarrow x$
			$v_1 \leftarrow y$
	otherwise		$v_0 \leftarrow y$
			$v_1 \leftarrow x$
			v

Результат

Результат в П-Ф оформлен в виде массива $\mathbf{v} = (v_0, v_1)$, так как по правилам описания имени П-Ф может быть присвоено значение **только одной переменной**

(в данном случае \mathbf{v} – это имя одной переменной-массива).

$$\begin{pmatrix} x \\ y \end{pmatrix} := P(4, 9) = \begin{pmatrix} 9 \\ 4 \end{pmatrix}$$

$$x = 9 \quad y = 4$$

Программирование циклических алгоритмов

По способам организации циклов в П-Ф выделяются две группы:

- а) *циклы типа арифметической прогрессии;*
- б) *итерационные циклы.*

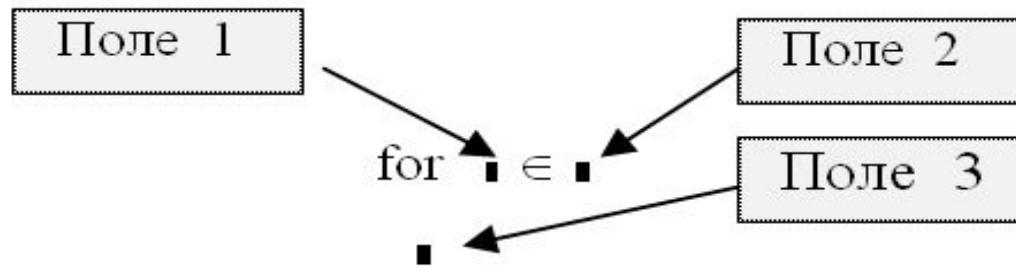
Программирование циклов типа арифметической прогрессии

Для программирования таких циклов используется оператор цикла *for* с параметром.

Особенности оператора цикла *for*

1. Параметр цикла может принимать значения различных типов: численные (целые или вещественные), текстовые и др.
2. Значения параметра цикла могут задаваться *дискретной* *переменной,*
последовательностью *чисел,* *массивом*

Структура оператора цикла *for*



Для ввода оператора цикла с параметром надо щелкнуть на кнопке ***for*** на **Панели программирования**.

В **Поле 1** вводится *имя переменной*, являющейся параметром цикла.

В **Поле 2** задается *закон изменения параметра* цикла.

В **Поле 3** вводятся операторы, составляющие тело цикла. Если одного поля недостаточно, то дополнительные поля для ввода операторов

Примеры программирования циклических алгоритмов с параметром цикла

Пример 1. Составить описание П-Ф, реализующей формирование вектора z из n ($n=5$) элементов, определяемых по правилу:

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

Решение

ORIGIN := 1

vect(n) := $\left| \begin{array}{l} \text{for } i \in 1..n \\ \\ z_i \leftarrow \frac{1}{i + 4} \\ \\ z \end{array} \right.$

$z := \text{vect}(5)$ $z^T = (0.2 \quad 0.167 \quad 0.143 \quad 0.125 \quad 0.111)$

Примеры программирования циклических алгоритмов с параметром цикла

Пример 2. Для переменной x ,
изменяющейся
от **0.5** до **1.5** с шагом **0.2** сформировать вектор
 q , состоящий из значений
функции:

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

где a и b – заданные вещественные числа.

Примеры программирования циклических алгоритмов с параметром цикла

Пример 2 (решение)

Параметр цикла – переменная i .

ORIGIN := 1

$x_n := 0.5$

$x_k := 1.5$

$d := 0.2$

m := $\text{trunc}\left(\frac{x_k - x_n}{d}\right) + 1 = 6$

$f(a,b) :=$

	for $i \in 1..m$
	$x \leftarrow x_n + (i-1) \cdot d$
	$y_i \leftarrow \frac{\ln(x)}{a^2 + b^2}$
	y

$q := f(1,2) =$

	-0.139
	-0.071
	-0.021
	0.019
	0.052
	0.081

Таблица базовых циклических алгоритмов

Алгоритм	Начальная установка (до цикла)	Основной оператор (в теле цикла)						
<p>Табулирование функции</p>	<p>Вывод заголовка таблицы</p> <table border="1" data-bbox="718 929 1149 1043"> <tr> <td>X</td> <td>Y</td> </tr> </table>	X	Y	<p>Вычисление и вывод</p> <table border="1" data-bbox="1306 836 1736 1058"> <tr> <td>1.0</td> <td>5.32</td> </tr> <tr> <td>...</td> <td>...</td> </tr> </table>	1.0	5.32
X	Y							
1.0	5.32							
...	...							
<p>Организация счетчика</p>	<table border="1" data-bbox="718 1143 1149 1258"> <tr> <td>$K \leftarrow 0$</td> </tr> </table>	$K \leftarrow 0$	<table border="1" data-bbox="1306 1143 1736 1258"> <tr> <td>$K \leftarrow K + 1$</td> </tr> </table>	$K \leftarrow K + 1$				
$K \leftarrow 0$								
$K \leftarrow K + 1$								

Таблица базовых циклических алгоритмов

Алгоритм	Начальная установка (до цикла)	Основной оператор (в теле цикла)
Накопление суммы	$S \leftarrow 0$	$S \leftarrow S + y$
Накопление произведения	$P \leftarrow 1$	$P \leftarrow P * y$

Таблица базовых циклических алгоритмов

Алгоритм	Начальная установка (до цикла)	Основной оператор (в теле цикла)
Поиск минимального значения	$\text{min} \leftarrow +10^6$	$\text{min} \leftarrow y \text{ if } y < \text{min}$
Поиск максимального значения	$\text{max} \leftarrow -10^6$	$\text{max} \leftarrow y \text{ if } y > \text{max}$