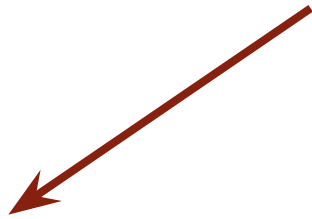


ПРОГРАММИРОВАНИЕ В MATHCAD

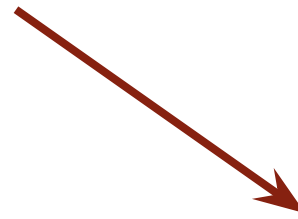


Программирование в MathCAD



безмодульное программирование

реализуется записью соответствующих конструкций непосредственно в математических областях документа MathCAD. Используется для сравнительно простых алгоритмов



модульное программирование

реализуется в виде отдельных независимых алгоритмов вычисления, представляющих собой отдельные программные модули (*подпрограммы -функции (П-Ф)*)



Безмодульное программирование в MathCAD

Программирование линейных алгоритмов

Особенность: строго последовательное выполнение всех операций алгоритма без пропусков и повторений вычислений.

Составить программу для вычисления корней квадратного уравнения

$$a := 2 \quad b := 5 \quad c := 8$$

$$d := b^2 - 4 \cdot a \cdot c$$

$$x_1 := \frac{-b + \sqrt{d}}{2 \cdot a}$$

$$x_2 := \frac{-b - \sqrt{d}}{2 \cdot a}$$

$$x_1 = -1.25 + 1.561i$$

$$x_2 = -1.25 - 1.561i$$

Проверка найденных корней:

$$a \cdot x_1^2 + b \cdot x_1 + c = 0$$

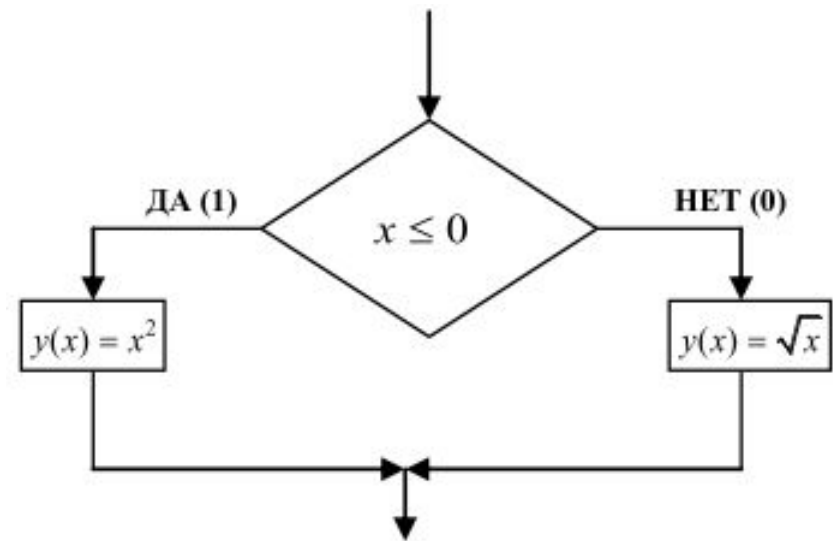
$$a \cdot x_2^2 + b \cdot x_2 + c = 0$$



Программирование разветвляющихся алгоритмов

Особенность: присутствие нескольких ветвей вычислительного процесса. Выбор конкретной ветви зависит от выполнения (или невыполнения) заданных условий.

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



Для проверки заданных условий в MathCAD используется:

- выражение отношений;
- логические операции;
- логические выражения.



Программирование разветвляющихся алгоритмов

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:

$\langle \mid \leq \mid \rangle \mid \geq \mid = \mid \neq$

$x := 3$	$y := -2$	$z := 6$
$x > y = 1$	$x - 2 \cdot y \geq 3 \cdot z = 0$	
$0 \leq x \leq 5 = 1$	$-4 \leq x - y \leq 3 = 0$	



Программирование разветвляющихся алгоритмов

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

NOT \neg	AND \wedge	OR \vee	XOR \oplus
$0\neg=1$	$0\wedge 0=0$	$0\vee 0=0$	$0\oplus 0=0$
$1\neg=0$	$0\wedge 1=0$	$0\vee 1=1$	$1\oplus 0=1$
	$1\wedge 0=0$	$1\vee 0=1$	$0\oplus 1=1$
	$1\wedge 1=1$	$1\vee 1=1$	$1\oplus 1=0$



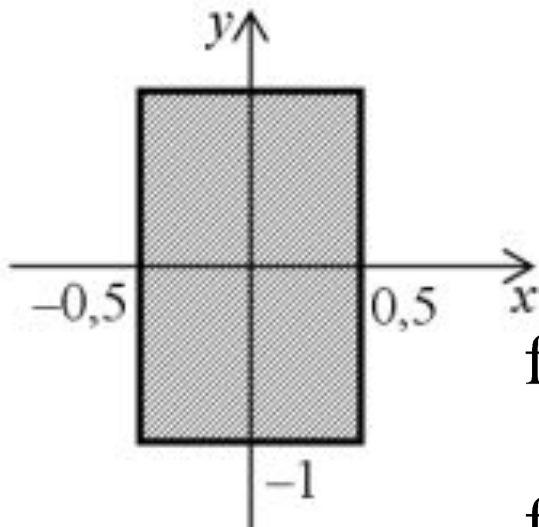
Программирование разветвляющихся алгоритмов

Логическое выражение – конструкция, состоящая из выражений отношений, логических операций и круглых скобок.

Принимает значения: 1 или 0.

Вычисляется слева направо с учетом приоритета входящих в выражение операций:

- круглые скобки;
- логическая операция И (AND);
- логические операции ИЛИ (OR) и исключающая ИЛИ (XOR);
- выражения отношений.



$$x := 0.2 \quad y := -0.8$$

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

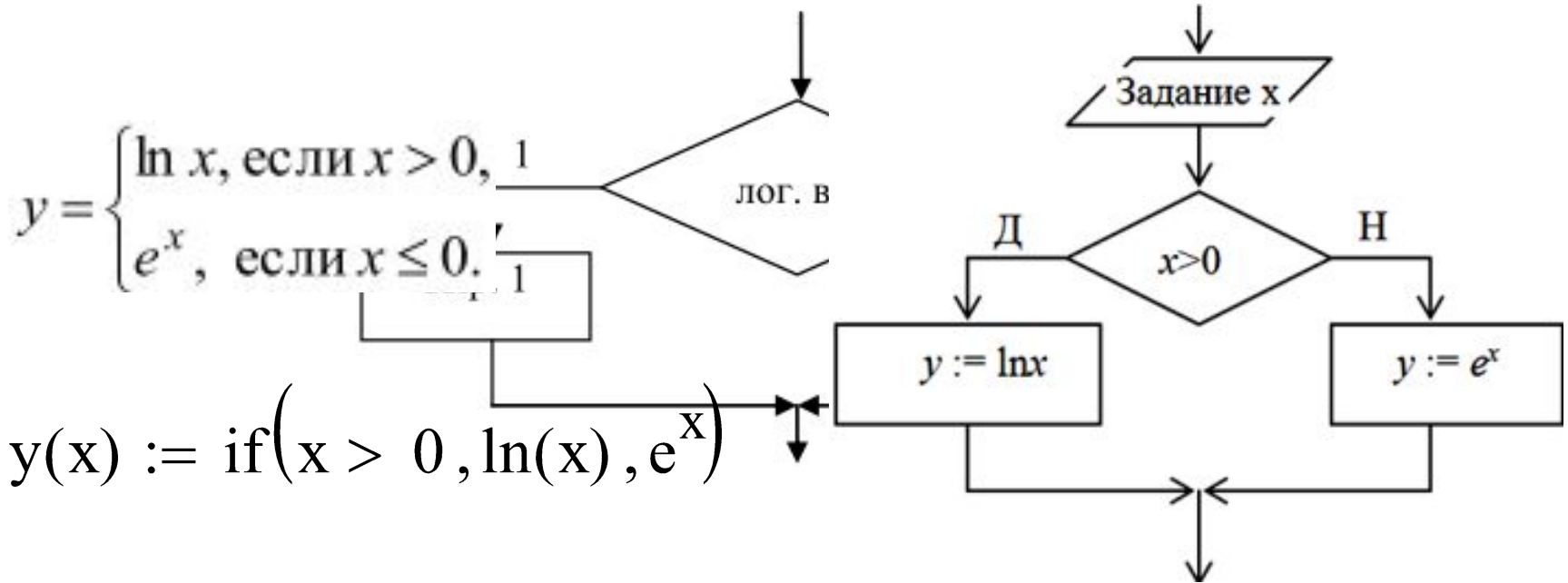
$$f(x, y) := (-0.5 \leq x \leq 0.5) \wedge (-1 \leq y \leq 1)$$

$$f(-1, 0.3) = 0$$

Условная функция *if*

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

где имя функции *if* вводится с клавиатуры.



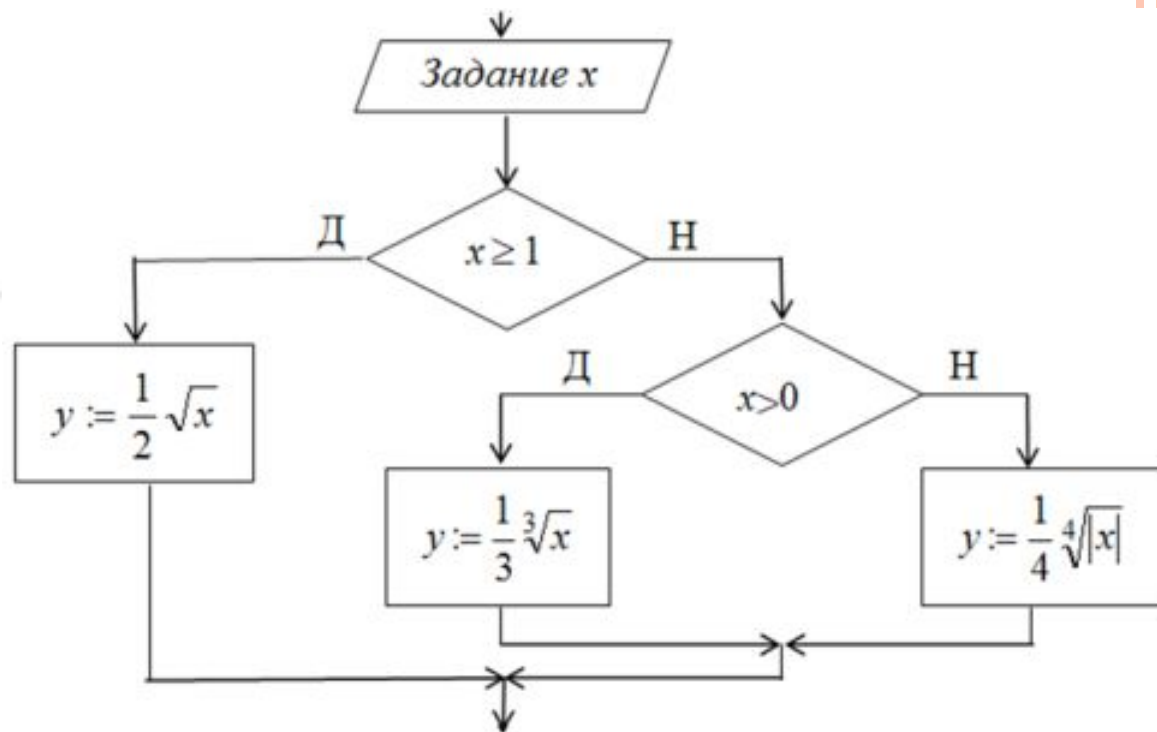
$$y(-3) = 0.05$$

$$y(2.71) = 0.997$$



Вложенная условная функция *if*

$$y = \begin{cases} \frac{1}{2} \sqrt{x}, & \text{если } x \geq 1, \\ \frac{1}{3} \sqrt[3]{x}, & \text{если } 0 < x < 1, \\ \frac{1}{4} \sqrt[4]{|x|}, & \text{если } x \leq 0. \end{cases}$$



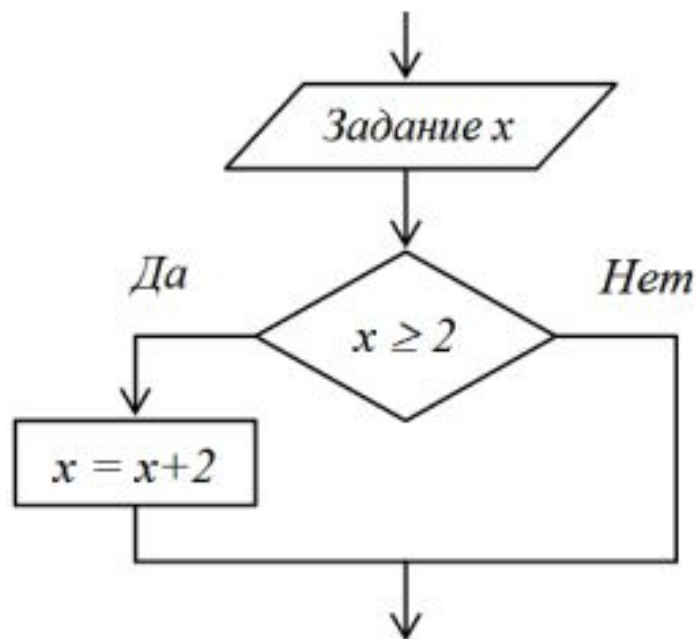
$$y(x) := \text{if} \left(x \geq 1, \frac{1}{2} \cdot \sqrt{x}, \text{if} \left(x > 0, \frac{1}{3} \cdot \sqrt[3]{x}, \frac{1}{4} \cdot \sqrt[4]{|x|} \right) \right)$$

$$y(2) = 0.707$$

$$y(0.5) = 0.265$$

$$y(-2) = 0.297$$

Запрограммировать алгоритм, вычисляющий величину x , по следующему правилу: если $x < 2$, то значение x оставить без изменения, в противном случае величину x увеличить на 2.



$$y(x) := \text{if}(x \geq 2, x + 2, x)$$

$$y(6) = 8$$

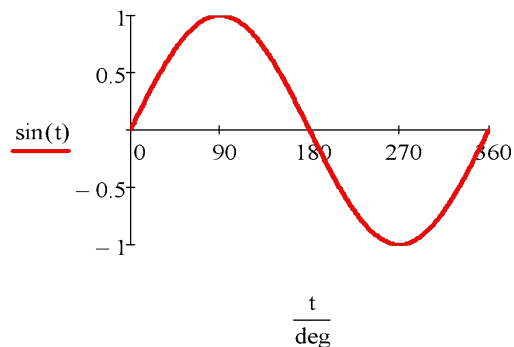
$$y(1) = 1$$



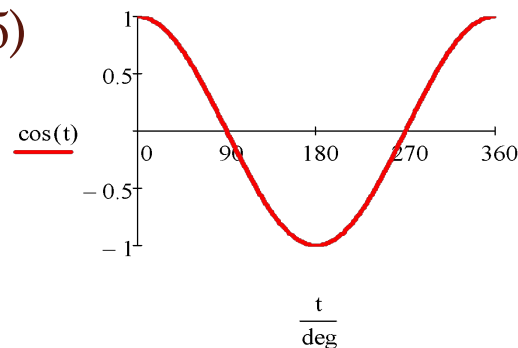
Составить функцию, осуществляющую построение графика в соответствии с рисунком (выполнить безмодульное программирование в MathCAD через условную функцию *if*).

Примеры графиков функций $f(t) = \text{Sin}(t)$ (рисунок а) и $f(t) = \text{Cos}(t)$ (рисунок б).

а)

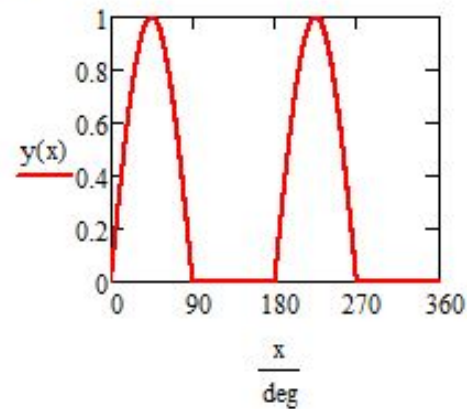


б)



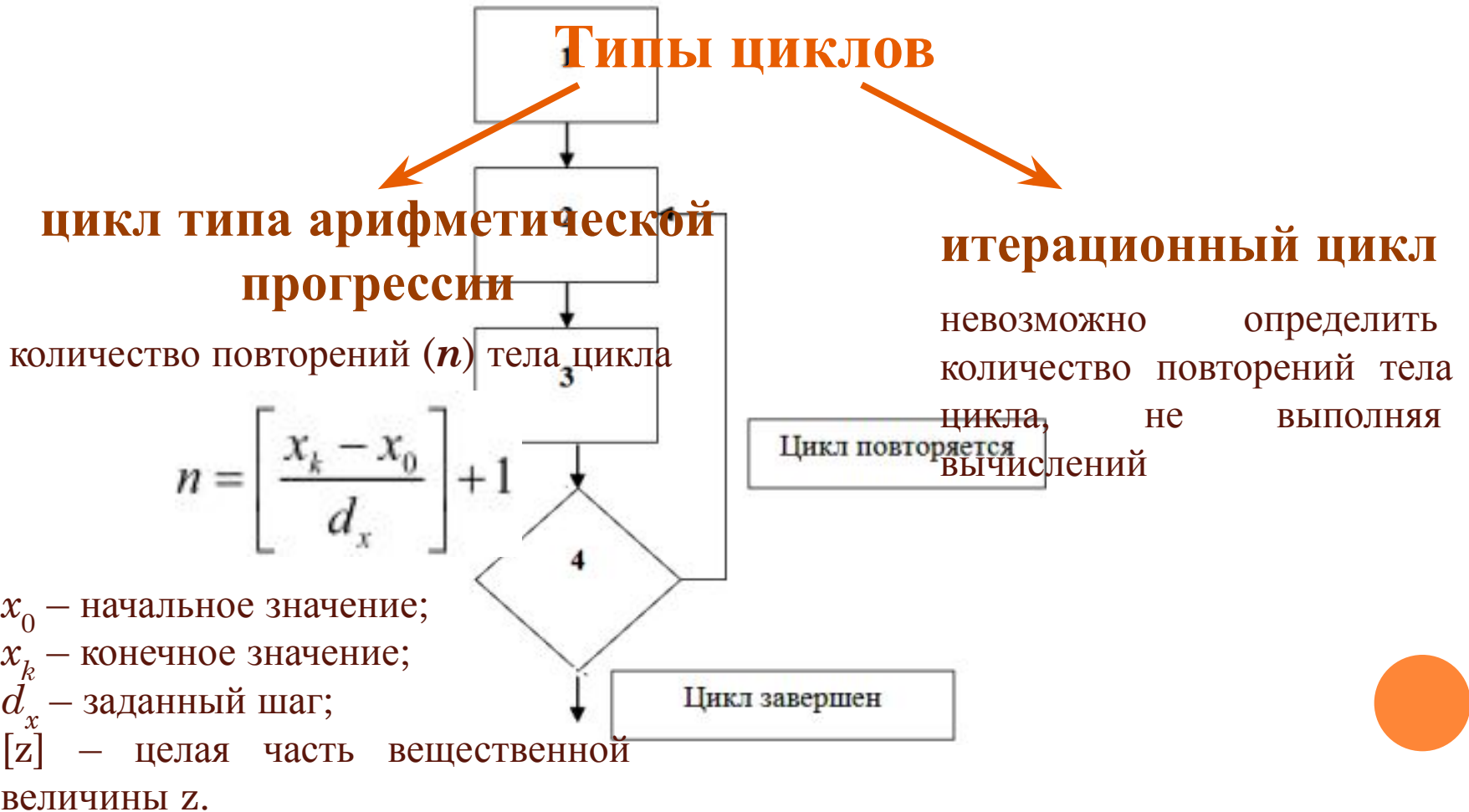
$$y(x) := \text{if} \left[\left(0 \leq x < \frac{\pi}{2} \right) \vee \left(\pi \leq x < \frac{3 \cdot \pi}{2} \right), \sin(2x), 0 \right]$$

$x := 0, 0.001 .. 2 \cdot \pi$



Программирование циклических алгоритмов

Особенность: содержит вычисления, повторяющиеся при различных значениях некоторой переменной, названной параметром цикла, а сами повторяющиеся вычисления составляют тело цикла.

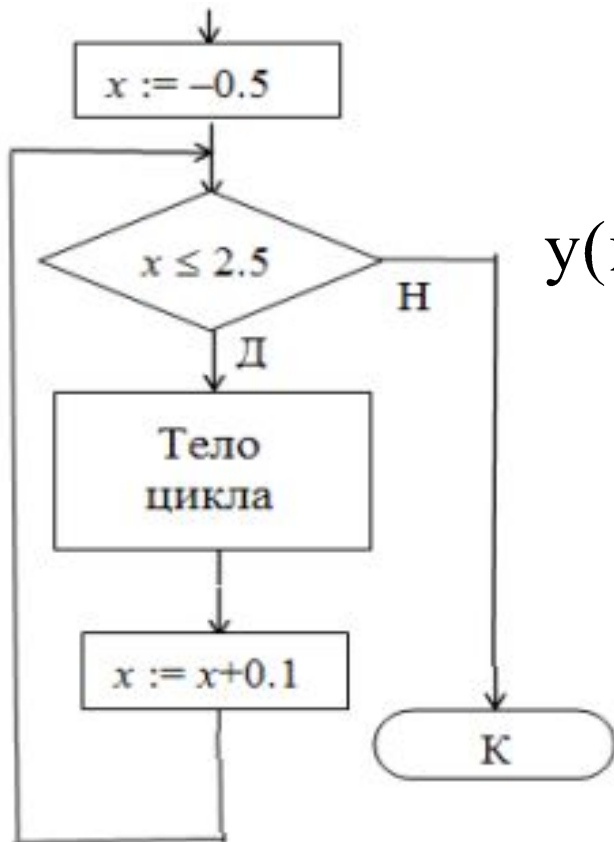


Цикл типа арифметической прогрессии

Вычислить значения функции

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

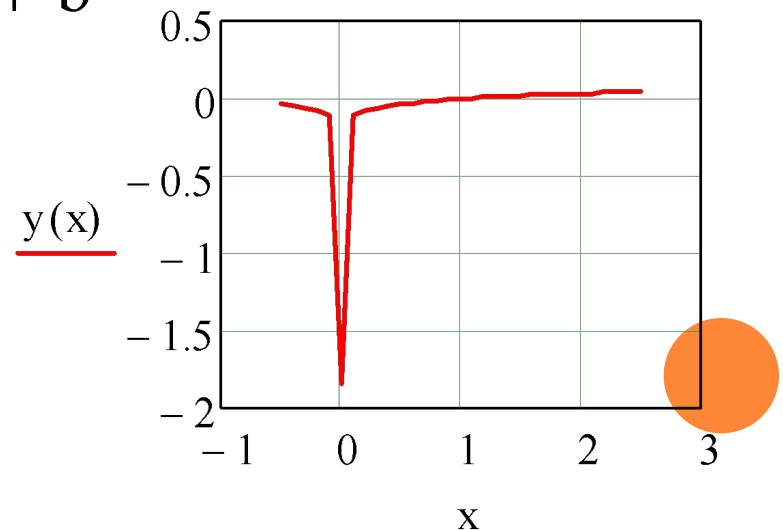
и построить график этой функции для всех x , изменяющихся в интервале $[-0.5, 2.5]$ с шагом $\Delta x = 0.1$; a, b – заданные вещественные числа.



$$a := 2 \quad b := 4$$

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

$$x := -0.5, -0.4 \dots 2.5$$



Цикл типа арифметической прогрессии

Сформировать матрицу (двумерный массив) B по следующему правилу:

$$B_{i,j} = \frac{1}{i+j+1}; i=1, \dots, 4; j=1, \dots, 6$$

ORIGIN := 1

n := 3

m := 6

i := 1 .. n

j := 1 .. m

$$B_{i,j} := \frac{1}{i+j+1}$$

$$B = \begin{pmatrix} 0.333 & 0.25 & 0.2 & 0.167 & 0.143 & 0.125 \\ 0.25 & 0.2 & 0.167 & 0.143 & 0.125 & 0.111 \\ 0.2 & 0.167 & 0.143 & 0.125 & 0.111 & 0.1 \end{pmatrix}$$



Итерационный цикл

Решить методом Ньютона

$$f(x) := 5 \cdot \sin(x) - 4 \cdot x^2 + 5.8$$

$$\varepsilon := 10^{-4}$$

$$x_0 := 3$$

$$n := 100 \quad i := 1..n \quad df(x) := \frac{d}{dx} f(x) \quad df(x) \rightarrow (5 \cdot \cos(3) - 24)$$

Вычисление первого приближения по формуле Ньютона

$$x_1 := x_0 - \frac{f(x_0)}{df(x_0)}$$

$$x_1 = 1.981$$

$$x_{i+1} := \text{until} \left(|x_i - x_{i-1}| - \varepsilon, x_i - \frac{f(x_i)}{df(x_i)} \right)$$

$$x^T = (3 \quad 1.9811938 \quad 1.6833021 \quad 1.6429796 \quad 1.6421985 \quad 1.6421982)$$

Определение числа итераций, за которые итерационный процесс сошелся:

$$j := \text{last}(x)$$

$$j = 5$$

