

## ЛАБОРАТОРНАЯ РАБОТА № 4

# РАБОТА С ФАЙЛАМИ, ИСПОЛЬЗОВАНИЕ УСЛОВНЫХ ФУНКЦИЙ И ПРОГРАММИРОВАНИЕ В СРЕДЕ MATHCAD

**Цель лабораторной работы** – ознакомиться с основными операторами, обеспечивающими работу с файлами данных, организацию вычислений с условными функциями, и изучить основы программирования в среде MathCAD.

### Работа с файлами данных

Для обмена с внешними программными системами MathCAD имеет специальный тип данных – *файловые данные*. В сущности, это те же векторы и матрицы, но с элементами, которые могут записываться в виде простых текстовых файлов. Ниже перечислены основные файловые операции.

**READPRN("Имя\_файла")**. Эта операция считывает данные в виде матрицы. Функция READPRN возвращает матрицу, значения элементов которой однозначно связаны со значениями элементов файла. Или, точнее, каждая строка или столбец возвращаемой матрицы подобны соответствующим строкам или столбцам текстового представления файла.

**WRITEPRN("Имя\_файла")**. Эта операция применяется для записи матричного выражения (или матрицы) в файл с указанным именем. Структура файла подобна структуре матрицы.

**APPENDPRN("Имя\_файла")**. Эта операция дописывает данные в уже существующий матричный файл. Следует особо отметить, что при работе с векторами и матрицами, имеющими комплексные элементы, эти операции используют расширенные векторы или матрицы, элементы которых – действительные числа.

Имя файла включает путь к файлу. Можно задавать как полный путь к файлу, например, C:\Мои документы, так и относительный, имея в виду, что он будет отсчитываться от папки, в которой находится файл с документом MathCAD. Если вы задаете в качестве аргумента просто имя файла, то файл будет записан или прочитан из той папки, в которой находится сам документ MathCAD.

На рис. 20 приведены примеры использования операторов для ввода-вывода данных.

## РАБОТА С ДАННЫМИ МАТРИЧНОГО ТИПА

$x := 0..4$	$y_x := x^2 - 10$	$y = \begin{pmatrix} -10 \\ -9 \\ -6 \\ -1 \\ 6 \end{pmatrix}$	Создание файла данных в корневом каталоге диска D
$\text{WRITEPRN} ("d:/dannie.prn" ) := y$	$r := \text{READPRN} ("d:/dannie.prn" )$	$r = \begin{pmatrix} -10 \\ -9 \\ -6 \\ -1 \\ 6 \end{pmatrix}$	Считывание файла данных, создание вектора r
$M := \begin{pmatrix} -5 & 7 \\ 2 & -9 \end{pmatrix}$	$N := \begin{pmatrix} 1 + 3i & 2 + 4i \\ 2 - 8i & 4 - 5i \end{pmatrix}$		Определение матриц
$\text{WRITEPRN} ("d:/matr.prn" ) := N$			Запись матрицы в файл
$\text{NN} := \text{READPRN} ("d:/matr.prn" )$			
$\text{MN} := \text{NN} + M$	$\text{MN} = \begin{pmatrix} -4 + 3i & 9 + 4i \\ 4 - 8i & -5 - 5i \end{pmatrix}$		Определение матрицы
$A := \begin{pmatrix} 5 - 2i & 5i \\ 2 + 6i & 1 + 3i \end{pmatrix}$			Дозапись матрицы A в существующий файл
$\text{APPENDPRN} ("d:/matr.prn" ) := A$	$\text{NA} := \text{READPRN} ("d:/matr.prn" )$	$\text{NA} = \begin{pmatrix} 1 + 3i & 2 + 4i \\ 2 - 8i & 4 - 5i \\ 5 - 2i & 5i \\ 2 + 6i & 1 + 3i \end{pmatrix}$	Считывание файла данных, создание матрицы

Рис. 20

## Использование функций с условиями сравнения

В системе MathCAD существует ряд встроенных функций, у которых возвращаемый ими результат зависит от знака или значения аргумента. Так, при их вычислении производится сравнение аргумента с некоторыми числовыми константами, например с нулем или целыми числами. Ниже представлены такие функции с условиями сравнения:

**ceil(x)** – наименьшее целое, большее или равное x;

**floor(x)** – наибольшее целое, меньшее или равное x;

**mod(x,y)** – остаток от деления x/y со знаком x;

**angle(x,y)** – положительный угол между осью x и радиус-вектором точки с координатами (x, y);

**Φ(x)** – функция Хевисайда – единичного скачка (дает 0 при  $x < 0$  и 1 в противном случае);

**δ(m,n)** – функция, именуемая символом Кронекера, возвращающая 1 при  $m = n$  и 0 в противном случае.

Функцию Хевисайда можно использовать для задания прямоугольного импульса с шириной  $\tau$ :

$$\text{pulse}(t, \tau) := \Phi(t) - \Phi(t - \tau).$$

Более широкие возможности дает функция **if** для создания условных выражений:

$$\text{if}(\text{Условие}, \text{Выражение 1}, \text{Выражение 2}).$$

Если в этой функции условие выполняется, то вычисляется выражение 1, в противном случае – выражение 2 (рис. 21).

## Использование программных модулей

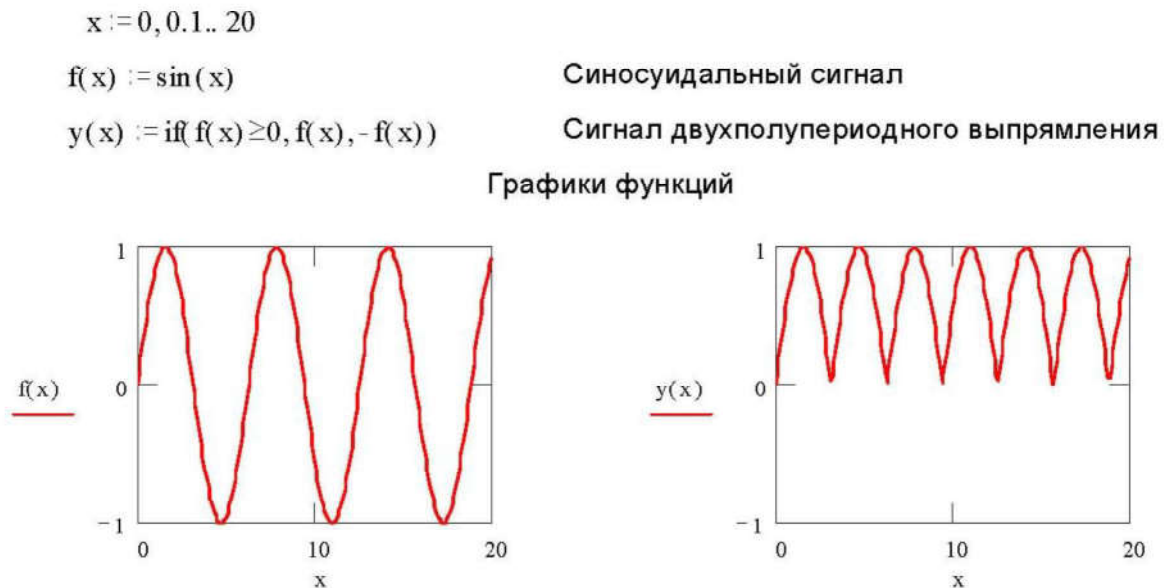


Рис. 21

Программные операторы сосредоточены в наборной панели программных элементов (рис. 22). Набор программных элементов весьма ограничен.

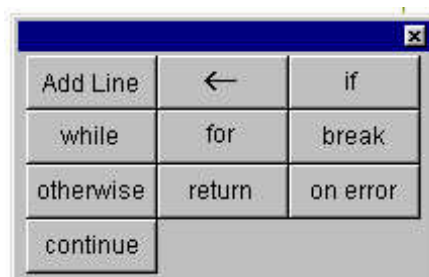


Рис. 22

**Add Line** – создает и при необходимости расширяет жирную вертикальную линию, справа от которой в шаблонах задается запись программного блока;

$\leftarrow$  – символ локального присваивания (телепрограммного модуля);

**if** – оператор условного выражения;

**for** – оператор задания цикла с фиксированным числом продолжений

**while** – оператор задания цикла типа "пока" (цикл выполняется, пока выполняется некоторое условие);

**otherwise** – оператор иного выбора (обычно применяется с **if**);

**break** – оператор прерывания;

**continue** – оператор продолжения;

**return** – оператор-функция возврата;

**on error** – оператор обработки ошибок.

Пример использования программных операторов приведен на следующей странице (рис. 23).

$x := 14$	Задано значение $x$
$\ln(x) = 2.639$	Вычислен натуральный логарифм
$\left  \begin{array}{l} x \leftarrow 5 \\ \ln(x) \end{array} \right. = 1.609$	Локально задано $x=5$ и вычислен натуральный логарифм от 5.
$x = 14$	За пределами программного блока $x$ сохранил значение 14.
$f(x, y, z) := \frac{x}{x+y \cdot z} + \frac{y}{x+y \cdot z} + \frac{z}{x+y \cdot z}$	Задание функции обычным способом
$f(5, 7, -3) = -0.563$ $f(2, 5, 7) = 0.378$	Вычисление значений функции
$Ff(x, y, z) := \left  \begin{array}{l} as \leftarrow x + y \cdot z \\ \frac{x + y + z}{as} \end{array} \right.$	Задание функции программным блоком
$Ff(5, 7, -3) = -0.563$ $Ff(2, 5, 7) = 0.378$	Вычисление значений функции

Рис. 23

**Оператор Add Line** выполняет функции расширения программного блока. Расширение фиксируется удлинением вертикальной черты программных блоков или их древовидным расширением.

**Оператор внутреннего присваивания**  $\leftarrow$  выполняет функции внутреннего локального присваивания. Например, выражение  $x \leftarrow 12$

присваивает переменной  $x$  значение 12. Локальный характер присваивания означает, что такое значение  $x$  сохраняет только в теле программы. За пределами тела программы значение переменной может быть не определенным либо равно значению, которое задается оператором локального  $:=$  и глобального  $\equiv$  присваивания.

**Оператор создания условных выражений if** задается в виде

Выражение if Условие

Если условие выполняется, то возвращается значение выражения. Совместно с этим оператором часто используются операторы прерывания **break** и оператор иного выбора **otherwise**.

**Оператор for** служит для организации циклов с заданным числом повторений. Он записывается в виде

for Var  $\in$  Nmin..Nmax

Эта запись означает, что если переменная Var меняется с шагом +1 от значения Nmin до Nmax, то выражение, помещенное в шаблон, будет выполняться. Переменную счетчика Var можно использовать в выражениях программы.

**Оператор While** служит для организации циклов, действующих до тех пор, пока выполняется некоторое условие. Этот оператор записывается в виде

while Условие

Выполняемое выражение записывается на место шаблона.

**Оператор otherwise** (иначе) обычно используется совместно с оператором if. Его использование поясняет следующая программная конструкция:

$f(x) := \begin{cases} 1 & \text{if } x > 0 \\ (-1) & \text{otherwise} \end{cases}$	Возвращает 1 если $x > 0$
	Возвращает -1 во всех иных случаях

**Оператор break** вызывает прерывание работы программы всякий раз, как он встречается. Чаще всего он используется совместно с оператором условного выражения if и операторами циклов **while** и **for**, обеспечивая переход в конец тела цикла.

**Оператор continue** (продолжения) используется для продолжения работы после прерывания программы. Он также используется обычно совместно с операторами задания циклов **while** и **for**, обеспечивая после прерывания возврат в начало цикла.

**Оператор-функция возврата return** прерывает выполнение программы и возвращает значение своего операнда, стоящего следом за ним. Например, в приведенном ниже случае

return 0 if  $x < 0$

будет возвращаться значение 0 при любом  $x < 0$ .

**Оператор on error и функция error.** Оператор обработки ошибок позволяет создавать конструкции обработчиков ошибок. Этот оператор задается в виде

Выражение\_1 on error Выражение\_2

Здесь если при выполнении Выражения\_1 возникает ошибка, то выполняется Выражение\_2. Для обработки ошибок полезна также функция **error(S)**, которая будучи в программном модуле возвращает окошко с надписью, хранящейся в символьной переменной *S* или в символьной константе (любой фразе в кавычках).

На следующей странице рассмотрены примеры использования программных блоков.

**Вложенные циклы** позволяют сделать программу компактней. Рассмотрен пример решения с использованием вложенных циклов следующей задачи. Дан массив чисел  $a_1..a_{10}$ . Вычислить  $a_1 + a_2^2 + \dots + a_{10}^{10}$  (рис. 24, 25).



Рис. 24



2. Выполнить средствами пакета MathCAD последовательность заданий из вариантов, указанных преподавателем, формируя единый документ. Каждую задачу обязательно сопровождать комментариями.

2.1. Выполнить задание 1.

2.2. Выполнить задание 2.

2.3. Используя условную функцию if, построить график ступенчатой функции задания 3.

2.4. Используя программный модуль решить задачу из задания 4.

2.5. Используя программный модуль решить задачу из задания 5.

2.6. Используя подпрограммы, решить задачу из задания 6.

### **Содержание отчета**

1. Титульный лист.

2. Решение всех задач с комментариями.

3. Выводы.

### **Контрольные вопросы**

1. Перечислите основные операторы для работы с файлами данных.

2. Напишите структуру оператора условия if.

3. Назовите основные программные операторы MathCAD.

4. Какие операторы позволяют организовать циклы с неизвестным количеством повторений ?

5. Какие операторы позволяют организовать циклы с известным количеством повторений?

6. Каким образом можно организовать вложенные циклы ?

## ЗАДАНИЯ ДЛЯ ВЫПОЛНЕНИЯ

**Задание 1.** Сформировать массив чисел по выражению, заданному в таблице. Записать значения элементов массива в файл с указанным именем. Сосчитать массив из записанного файла и умножить все его элементы на указанный скаляр.

Номер варианта	Длина вектора	Выражение	Скаляр	Имя файла
1	8	$\frac{i}{i+1}$	2,5	first
2	7	$\frac{1}{i+1}$	-2,7	second
3	9	$\frac{5}{i}$	3,6	third
4	8	$\frac{10}{i+1}$	3,2	fourth
5	7	$\frac{1}{2i}$	-4,3	fifth
6	9	$\frac{1}{2i+1}$	2,8	sixth
7	8	$\frac{5}{2i+1}$	-1.7	seventh
8	7	$\frac{3}{0,5i+1}$	3,9	eighth
9	9	$\frac{1}{0,3i+3}$	-2,4	ninth
10	6	$\frac{1}{0,7i+1}$	3,5	tenth

**Задание 2.** Сформировать матрицу размером и по выражениям, заданным в таблице. Записать значения элементов этой матрицы в файл. Найти максимальный элемент. Обнулить матрицу путем умножения на 0. Сосчитать значения матрицы из файла и вывести ее на экран.

Номер варианта	Размер матрицы	Выражение	Имя файла
1	4×4	$a_{i,j} = (i+1) + (j-1)$	prima
2	4×5	$a_{i,j} = i \cdot j - i$	seconda
3	3×5	$a_{i,j} = 2 \cdot i + j$	tercia
4	5×3	$a_{i,j} = (i+j) + (i-j)$	quarta
5	3×3	$a_{i,j} = i + j / 2$	quinta
6	4×3	$a_{i,j} = i + j / 3 - 1$	sexta
7	3×4	$a_{i,j} = (i+j)/5 + 2$	septima
8	4×4	$a_{i,j} = (i+j)/4 - 2$	octava
9	5×4	$a_{i,j} = (i+j)/7 + 2$	nonan
10	5×5	$a_{i,j} = (i+j)/3 + 1$	desima

### Задание 3.

$$1. y(x) = \begin{cases} -x & \text{при } x < 0 \\ -x^2 & \text{при } x \geq 0 \end{cases}$$

$$2. y(x) = \begin{cases} \frac{1}{x^2} & \text{при } x < -1 \\ x^2 & \text{при } -1 \leq x \leq 2 \\ 4 & \text{при } x > 2 \end{cases}$$

$$3. y(x) = \begin{cases} x^2 & \text{при } x < 0 \\ -x^4 & \text{при } x \geq 0 \end{cases}$$

$$4. y(x) = \begin{cases} -x-1 & \text{при } x < -1 \\ x+1 & \text{при } -1 \leq x \leq -0 \\ -x+1 & \text{при } -0 \leq x \leq 1 \\ x-1 & \text{при } x > 1 \end{cases}$$

$$5. y(x) = \begin{cases} -3x-1 & \text{при } x \leq -2 \\ -x+6 & \text{при } -2 \leq x \leq 1 \\ x+4 & \text{при } 1 \leq x \leq 3 \\ 3 \blacksquare - 2 & \text{при } x > 3 \end{cases}$$

$$6. y(x) = \begin{cases} -1 & \text{при } \blacksquare < -1 \\ x & \text{при } -1 \leq x \leq 1 \\ 1 & \text{при } x > 1 \end{cases}$$

$$7. y(x) = \begin{cases} -x^2 & \text{при } x < 0 \\ -x^3 & \text{при } x \geq 0 \end{cases}$$

$$8. y(x) = \begin{cases} -x & \text{при } x < -1 \\ 1 & \text{при } -1 \leq x \leq 1 \\ x & \text{при } x > 1 \end{cases}$$

$$9. y(x) = \begin{cases} -\frac{1}{x^2} + 1 & \text{при } x < -1 \\ x^2 + 1 & \text{при } -1 \leq x \leq 2 \\ 5 & \text{при } x > 2 \end{cases}$$

$$10. y(x) = \begin{cases} x+1 & \text{при } x < -1 \\ -x-1 & \text{при } -1 \leq x \leq -0 \\ x-1 & \text{при } -0 \leq x \leq 1 \\ -x+1 & \text{при } x > 1 \end{cases}$$

#### Задание 4.

1. Вычислить  $\sum_{i=1}^{100} \frac{1}{i^2}$

2. Вычислить  $\sum_{i=1}^{50} \frac{1}{i^3}$

3. Вычислить  $\sum_{i=1}^{128} \frac{1}{i^3}$

4. Вычислить  $\sum_{i=1}^{120} \frac{1}{i}$

5. Вычислить  $\sum_{i=1}^9 \frac{1}{i^5}$

6. Вычислить  $\sum_{i=1}^{39} \frac{1}{(2i+1)^2}$

7. Вычислить  $\sum_{i=1}^{12} \frac{(-1)^{i+1}}{i(i+1)}$

8. Вычислить  $\sum_{i=1}^{25} \frac{(-1)^i}{i(2i+1)}$

9. Вычислить  $\sum_{i=1}^{90} \frac{(-1)^{i+1}}{i(i+1)(i+2)}$

10. Вычислить  $\sum_{i=1}^5 \frac{(-1)^{i+1}}{4^i + 5^{i+2}}$

**Задание 5.** Вычислить сумму ряда с заданной точностью  $\varepsilon$ . Считать, что требуемая точность достигнута, если вычислена сумма нескольких первых слагаемых и очередное слагаемое оказалось по модулю меньше, чем  $\varepsilon$ . Указать количество учтенных слагаемых.

1. Для  $x = 1,28$  и  $\varepsilon = 0,000001$   $\sum_{k=1}^{\infty} \frac{1}{x^3 k^2}$ .

2. Для  $x = 3,51$  и  $\varepsilon = 0,000001$   $\sum_{k=1}^{\infty} \frac{1}{x^2 + k^3}$ .

3. Для  $x = 1,21$  и  $\varepsilon = 0,000001$   $\sum_{k=1}^{\infty} \frac{x^2}{k^{3/2}}$ .

4. Для  $x = 2,47$  и  $\varepsilon = 0,000001$   $\sum_{k=1}^{\infty} \frac{1}{\sqrt{|x|} + k^2}$ .

5. Для  $x = 3,11$  и  $\varepsilon = 0,000001$   $\sum_{k=1}^{\infty} \frac{\sqrt{|x|}}{k^3}$ .

6. Для  $x = 1,85$  и  $\varepsilon = 0,000001$   $\sum_{k=1}^{\infty} \frac{x}{k^3 + k\sqrt{|x|} + 1}$ .

7. Для  $x = 2,01$  и  $\varepsilon = 0,000001$   $\sum_{k=1}^{\infty} \frac{(-1)^k x^k}{k}$ .

8. Для  $x = 1,09$  и  $\varepsilon = 0,000001$   $\sum_{k=1}^{\infty} \frac{(-1)^k (\lfloor x \rfloor + 1)x^k}{3^k}$ .

9. Для  $x = 3,12$  и  $\varepsilon = 0,000001$   $\sum_{k=1}^{\infty} \frac{(-1)^k x^k}{(\lfloor x \rfloor + 1)^2}$ .

10. Для  $x = 0,89$  и  $\varepsilon = 0,000001$   $\sum_{k=1}^{\infty} x^{\lfloor x^2 \rfloor}$ .

### Задание 6.

Номер варианта	Выражение	Номер варианта	Выражение
1	$\sum_{i=1}^{100} \sum_{j=1}^{50} \frac{1}{i+j^2}$	6	$\sum_{i=1}^{50} \sum_{j=1}^{10} \cos(i^2 + j^3)$
2	$\sum_{i=1}^{100} \sum_{j=1}^{60} \sin(i^3 + j^4)$	7	$\sum_{i=1}^{100} \sum_{j=1}^{100} \frac{j+i-1}{2i+j}$
3	$\sum_{i=1}^{100} \sum_{j=1}^{100} \frac{j-i+1}{i+j}$	8	$\sum_{i=1}^{50} \sum_{j=1}^{60} \sin(i^2 + j^3)$
4	$\sum_{i=1}^{100} \sum_{j=1}^i \frac{1}{i+2j}$	9	$\sum_{i=1}^{20} \sum_{j=1}^{20} 5\cos(i^3 + j^2)$
5	$\sum_{i=1}^{10} i^3 \sum_{j=1}^{15} (i-j)^2$	10	$\sum_{i=1}^{30} \sum_{j=i}^{20} \frac{2x+i}{0,5j} \text{ для } x = 3,2$