

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ
АРХИТЕКТУРНО-СТРОИТЕЛЬНЫЙ УНИВЕРСИТЕТ
(СИБСТРИН)

Кафедра прикладной математики

**ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ
(Лабораторный практикум
в Mathcad)**

Лабораторные работы
и методические указания по их выполнению
для студентов, обучающихся
по направлению подготовки 08.03.01 «Строительство»
и специальности 08.05.01 «Строительство
уникальных зданий и сооружений»
всех форм обучения

НОВОСИБИРСК 2019

Лабораторные работы и методические указания по их выполнению составлены канд. техн. наук, доцентом И.Н. Мухиной

Утверждены учебно-методической комиссией
факультета вечернего и заочного обучения
15 мая 2019 года

Рецензенты:

- Ю.Е. Воскобойников, д-р физ.-мат. наук,
профессор кафедры прикладной математики
НГАСУ (Сибстрин);
- А.Ф. Задорожный, канд. техн. наук,
профессор кафедры информационных систем
и технологий НГАСУ (Сибстрин)

© Новосибирский государственный
архитектурно-строительный
университет (Сибстрин), 2019

СОДЕРЖАНИЕ

Лабораторная работа № 1. Интерфейс Mathcad и основы вычислений	2
Лабораторная работа № 2. Операторы математического анализа. Символьные вычисления	7
Лабораторная работа № 3. Обработка массивов. Работа с файлами	9
Лабораторная работа № 4. Построение графиков	14
Лабораторная работа № 5. Программирование линейных и разветвляющихся алгоритмов	16
Лабораторная работа № 6. Программирование простых циклов	20
Лабораторная работа № 7. Программирование двойных циклов	30
Лабораторная работа № 8. Решение нелинейных алгебраических уравнений	38
Лабораторная работа № 9. Решение систем линейных алгебраических уравнений	46
Лабораторная работа № 10. Часть I. Интерполирование функций	54
Лабораторная работа № 10. Часть II. Аппроксимация функций	61
Лабораторная работа № 11. Численное интегрирование	64
Лабораторная работа № 12. Численное решение обыкновенных дифференциальных уравнений (задача Коши)	68
Контрольная работа	73
Список литературы	75

Лабораторная работа № 1

Интерфейс Mathcad и основы вычислений

Основными компонентами интерфейса математического пакета Mathcad являются: строка главного меню, панели быстрых кнопок, рабочая область, палитры инструментов.

Многооконный режим работы. Mathcad поддерживает *многооконный* режим работы, т.е. предоставляет возможность одновременной работы с несколькими документами.

Задание 1

- Загрузить Mathcad, открыть окно программы на весь экран.
- Сохранить файл с новым именем.
- Создать второй документ и расположить его с первым один над другим, один бок о бок с другим, каскадом.

Текстовая область. Рабочая область предназначена для выполнения расчетов, здесь же можно вводить текст. Текстовая область – это зона ввода любого текста в сплошной рамке с *маркерами*.

Задание 2

- Щелкнуть мышью в любом месте рабочей области.
- Используя команды главного меню *Insert – Text Region*, ввести любой текст из нескольких строк.
- Переместить текстовую зону.
- Скопировать текстовую зону.
- Изменить размер, цвет, жирность текста в копии.

Справочная система

Задание 3

- Используя команды главного меню *Help – Mathcad Help*, найти названия функций, выполняющих *сортировку*.
- По ссылке *QuickSheet* перейти на страницу помощи с примерами применения этих функций.

Палитры инструментов содержат быстрые кнопки для вызова сходных по назначению функций. Найти нужную палитру можно, используя команды главного меню *View – Toolbars*.

Область вычислений ограничена сплошной рамкой *без маркеров* и служит для ввода формул, их редактирования и вывода результатов расчетов. Для выполнения простейших расчетов понадобятся идентификаторы, константы и переменные.

Идентификаторы (имена переменных, функций) состояются из латинских и греческих букв, цифр, символов, знаков подчеркивания и могут иметь любую длину. Прописные и строчные буквы в идентификаторах *различаются*. Начинать имя должно только с *буквы*. Применение в имени русских букв и пробелов *запрещено*.

Для ввода *нижнего индекса* после имени необходимо набрать *точку*, например: $x.2$

Задание 4

В области вычислений документа ввести следующие идентификаторы: x_2 $x_2 \lambda$ Δ_i S_t

Константы. В Mathcad есть следующие типы констант:

- **целочисленные** (например, 3 -1000 +55);
- **вещественные** или **действительные**, где разделителем целой и дробной части является *точка* (например, -3.75). Экспоненциальная (научная) форма записи действительных чисел представляет числа в виде формулы $a \cdot 10^p$, где a – мантисса числа (обычно $0 < a < 1$), p – порядок (например, 0.135E+02 -0.48E-01);
- **комплексные** константы вида $c = a + i \cdot b$ (a – действительная часть комплексного числа c ; b – мнимая часть; i – обозначение мнимой единицы $i = \sqrt{-1}$; между i и мнимой частью *знака умножения НЕТ*. Например, -5.23+0.123i);
- **строковые** константы, представляющие собой последовательность любых символов, заключенную в апострофы (например, ‘Это строковая константа’);

- **системные** константы, хранящие значения параметров системы и физических величин (например, c – константа скорости света, g – ускорение свободного падения);
- **единицы измерения** физических величин (например, kg, cm, J).

Задание 5

В области вычислений документа ввести следующие константы: 0.15 -2.175 $0.43 \cdot 10^{-2}$ $2+7.5i$

Переменные. В Mathcad используются следующие типы переменных:

- **простые** (хранят одно значение);
- **дискретные** (хранят набор значений);
- **размерные** (физические величины).

Все вычисления выполняются сверху вниз и слева направо!

Использование **незаданной** переменной в вычисляемом выражении является ошибочным. Такую переменную Mathcad выделяет красным цветом. Значение переменной можно задать, используя **оператор присваивания**, имеющий следующий вид:

<Имя переменной > := <Выражение>

Для ввода знака := нажать клавишу [=] или значок с палитры *Calculator*. Например, $x := 2$

Оператор вывода. Для вывода на экран значения переменной или выражения достаточно после выражения нажать клавишу [=]. Тогда на экране после знака = выведется **численное** значение.

Можно в одном выражении выполнить оператор присваивания и оператор вывода, например: $y := 25^{1/2} = 5$

Задание 6

- Вывести значения системных констант g и c .
- Присвоить значения переменным:
 $a := 3.1$ $b_0 := 1.2 \cdot 10^{-4}$ $b_1 := 0.0023$ $b_2 := 2.14 \cdot 10^6$ $c := -4.678$

- Вычислить значения арифметических выражений.

Указание: использовать клавишу [пробел] для выделения **синим уголком** части формулы, к которой применяется следующая операция [1, § 2.2]:

$$a) \sqrt{c} \quad b) e^{-a+b_0} + \sqrt{b_0 + 2 \cdot b_1 - 0.001 \cdot b_2} \quad c) \sqrt{\frac{c + a \cdot e^{-a^2+2} + 7}{\left| a + \frac{1}{1+a^2} \right|}}$$

- Вычислить значение переменной y по формуле $d)$ и переменной z по формуле $e)$

$$d) \frac{1}{1 + \frac{1}{1+a} + \frac{1}{1+c}} + 1 + \frac{1+c}{a} \quad e) \frac{e^{-a} + e^{-c}}{\cos(a^2) + \sin(a)^2} + \ln|a-c|$$

Дискретная (ранжированная) переменная – переменная, принимающая ряд значений, меняющихся по закону арифметической прогрессии [1, § 2.1].

Задание 7

Задать и вывести значения следующих переменных:

- переменная i меняется от -3 до 5 с шагом 1
- переменная x меняется от 0.3 до 1.8 с шагом 0.1
- переменная z меняется от -4 до -2 с шагом 0.2

Линейные алгоритмы – это алгоритмы вычислений, в которых ни одно действие не пропускается, и ни одно действие не повторяется.

Задание 8

Открыть **НОВЫЙ** документ и запрограммировать в виде линейного алгоритма вычисление корней квадратного уравнения. Выполнить проверку найденных корней. Коэффициенты уравнения задать так, чтобы были получены вещественные корни и мнимые корни [1, § 7.1].

Функции. В состав Mathcad входит большая библиотека *встроенных* функций. Зайти в библиотеку функций можно командами главного меню *Insert – Function* или с помощью быстрой кнопки со значком $f(x)$ [1, § 2.3, рис. 2.4].

Наряду с использованием встроенных библиотечных функций, пользователь Mathcad имеет возможность создавать свои функции. Такие функции будем называть *функциями пользователя* [1, § 2.3, пример 2.3.1].

Задание 9

- Рассчитать значение арктангенса числа π .
- Рассчитать логарифм числа 4 по основанию 3.
- Для **задания 8** написать функцию вычисления детерминанта, вызвать ее в вычислениях корней уравнения.

Размерные переменные – это числовые переменные и функции, обладающие размерностью, содержащие значения в системе измерения СИ физических величин [1, § 2.1, рис. 2.1, пример 2.2.2].

Задание 10

- Задать значения массы и высоты: $m := 5.67 \text{ kg}$ $h := 120 \text{ cm}$
- Вывести константу ускорения свободного падения g
- Рассчитать потенциальную энергию: $E := m \cdot g \cdot h = 66.724 \text{ J}$

Лабораторная работа № 2
Операторы математического анализа.
Символьные вычисления

Используя палитру *Calculus*, выполнить следующие задания (примеры и дополнительную информацию см. в [1, § 2.4]).

1. Вычисление суммы и произведения. Вычислить:

а) сумму всех целых чисел от 1 до 100;

б) сумму ряда $n^2 \cdot \sin(n)$, где $n = 1, 2, \dots, 20$;

в) сумму всех значений $\frac{i + j^2}{2 \cdot i \cdot j}$, где $i = 1, 2, \dots, 10$

и $j = 1, 2, \dots, 5$;

г) факториал числа n ($n < 10$). Напомним, что $n! = 1 \cdot 2 \cdot \dots \cdot n$.

Сравнить со значением функции $n!$ с палитры *Calculator*;

д) произведение всех значений $\cos(2k - 1)$, где $k = 1, 2, \dots, 5$.

2. Вычисление производных в точке. Вычислить:

а) первую и вторую производную в точке $x = 3$ для функции $f(x) = 2x^5 - 7x^4 + x$;

б) первую производную в точках $x = 2$ и $x = 3$ для функции $g(x) = \sqrt{5 - x^2} + e^{x+1}$. В каком случае результатом будет действительное число, а в каком – комплексное число?

3. Вычисление определенных интегралов. Вычислить:

а) площадь под кривой $\sin(x^2) + \cos(\sqrt{x})$ для $x \in [0, \pi]$;

б) $\int_0^{\pi/4} \sin(x^4) dx$;

в) двойной определенный интеграл для функции $\sqrt{x^2 + y^2}$, $x = 0 \dots 1$, $y = 0 \dots x$.

Создайте НОВЫЙ документ. Используя палитру *Symbolic*, выполните следующие задания (примеры см. в [1, глава 5]).

4. Вычисление производных, неопределенных интегралов и пределов. Для функции $f(x) = \frac{\pi}{2} - x - \frac{x^3}{6} - \frac{3x^5}{40}$ найти:

- а) **уравнения** первой и второй производной;
- б) неопределенный интеграл;
- в) пределы при $x \rightarrow +\infty$ и $x \rightarrow -\infty$.

5. Аналитическое решение уравнений. Решить *аналитически* уравнения, используя оператор символьного вывода [\rightarrow]:

$$a) \int 2 \cdot \pi \cdot (x^2 - 1) dx \quad b) \int_{-\infty}^0 e^x \cdot x^2 dx \quad c) \sum_{i=0}^{\infty} e^{\frac{-i}{2}}$$

$$d) \lim_{x \rightarrow \pi} \frac{\cos(5x) - \cos(3x)}{\sin(x)^2} \quad e) \sqrt{1 + \left(\frac{d}{dx} (x^2 - 1) \right)^2}$$

Для уравнения *a)* проверить правильность вычисления, продифференцировав первообразную.

6. Символьные функции. Выполнить задания, используя символьные функции с палитры *Symbolic*:

- а) командой *Simplify* упростить выражение $\sum_{i=0}^n i^2$;
- б) командой *Expand* раскрыть скобки $(x-y) \cdot (x-z) \cdot (y-z)$;
- в) полученное выражение разложить на множители, используя команду *Factor*;
- г) командой *Coeffs* найти коэффициенты при переменных x и y для выражения $x^2 \cdot \sin(y) - x + y \cdot z$.

7. Разложение в ряд Тейлора. Разложить в ряд Тейлора:

- а) функцию $f(x) = 1 - \sin(x)$ с точностью до 5 и 7 степеней;
- б) функцию арктангенса с точностью до 3 степени;
- в) функцию котангенса с точностью до 7 степени.

Лабораторная работа № 3 Обработка массивов. Работа с файлами

Перед выполнением лабораторной работы необходимо вывести на экран значение системной переменной ORIGIN (записывается прописными буквами). По умолчанию $ORIGIN = 0$. Это означает, что нумерация элементов массивов начинается с нуля.

Для того, чтобы первый элемент массива имел номер (индекс) 1, необходимо изменить значение системной переменной в операторе присваивания $ORIGIN := 1$ или через команды главного меню *Tools – Worksheet Options...* (примеры и дополнительную информацию см. в [1, глава 3]).

Часть I. Одномерные массивы (векторы)

Для обработки одномерных массивов в Mathcad необходимо ввести дискретную переменную, которая будет означать порядковый номер (индекс) элемента в векторе: $i := 1..n$. Здесь n – размерность вектора.

Для обращения к одному элементу вектора используется индексная переменная x_i . Для ввода индексной переменной нажать кнопку x_n на палитре *Matrix*.

Задача 1

а) С помощью функции $runif(n, a, b)$ сформировать одномерный массив (вектор) x , состоящий из n действительных случайных чисел из диапазона (a, b) . Значения n ($n > 5$), a , b задать самостоятельно. Вывести вектор x на экран. Например:

$$n := 6 \quad a := -5 \quad b := 5 \quad x := runif(n, a, b) \quad x = \dots$$

б) Сформировать вектор y из элементов вектора x , округленных до целого значения. Вывести вектор y на экран. Для округления действительного числа использовать любую из функций $round(x)$, $trunc(x)$, $ceil(x)$, $floor(x)$. Например:

$$y := round(x) \quad y = \dots$$

в) Вычислить скалярное произведение векторов x и y , используя операторы палитры *Calculus*. Выполнить эту же опера-

цию, используя функцию скалярного произведения $\bar{x} \cdot \bar{y}$ палитры *Matrix*. Сравнить полученные результаты.

з) Вычислить длину вектора x , используя операторы палитры *Calculus* и функцию $|x|$ палитры *Matrix* (*Calculator*) [1, табл. 3.5]. Сравнить полученные результаты.

Задача 2

а) С помощью функции $rnorm(n,m,d)$ сформировать вектор x , содержащий значения случайной величины с нормальным распределением. Здесь n – количество элементов в векторе, m – математическое ожидание, d – стандартное отклонение случайной величины. Например:

$$n:=100 \quad m:=0 \quad d:=0.1 \quad x:=rnorm(n,m,d) \quad x=...$$

б) Вычислить выборочное среднее и выборочную дисперсию по формулам

$$mx = \frac{1}{n} \sum_{i=1}^n x_i; \quad Dx = \frac{1}{n} \sum_{i=1}^n (x_i - mx)^2.$$

в) Сравнить полученные значения с результатами расчетов стандартных функций $mean(x)$ и $var(x)$.

Часть II. Двумерные массивы (матрицы)

Для обработки двумерных массивов в Mathcad необходимо ввести две дискретные переменные, первая из которых означает номер строки, а вторая – номер столбца элемента матрицы: $i:=1..n$; $j:=1..m$. Здесь n, m – размерность матрицы.

Для обращения к одному элементу матрицы используется индексная переменная $A_{i,j}$.

Для ввода индексной переменной нажать кнопку x_n на палитре *Matrix*.

Задача 3

Сформировать следующие матрицы:

$$A = \begin{pmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & -1 \end{pmatrix} \quad B = \begin{pmatrix} 2 & 0 & -1 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix} \quad C = \begin{pmatrix} 2 & 8 & 3 \\ 3 & 0 & -2 \\ 7 & 2 & 1 \end{pmatrix}$$

Проверить, что матрицы A и B – перестановочные, т.е. $A \cdot B = B \cdot A$, а матрицы A и C – неперестановочные. Для этого вычислить матрицу $D = A \cdot B - B \cdot A$, вывести ее элементы и найти евклидову норму [1, табл. 3.4]. Повторить действия для матриц A и C .

Задача 4

а) Сформировать *квадратную* матрицу по формуле $A_{i,j} = \text{tg}(i^2 + j^2)$. Размерность матрицы задать самостоятельно.

б) Используя палитру *Matrix*, вычислить матрицу, обратную к матрице A [1, табл. 3.5]. Проверить правильность вычисления.

в) Используя палитру *Matrix*, вычислить транспонированную матрицу для матрицы A . Вычислить произведение исходной матрицы на транспонированную. Вычислить определитель и обратную матрицу для произведения матриц AA^T .

Задача 5

Дана система уравнений

$$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 30 \\ -x_1 + 2x_2 - 3x_3 + 4x_4 = 10 \\ x_2 - x_3 + x_4 = 3 \\ x_1 + x_2 + x_3 + x_4 = 10 \end{cases}$$

Найти решение этой системы, используя *обратную матрицу* и функцию *lsolve*. Вычислить вектор невязки [1, пример 3.5.2].

Задача 6

В **НОВОМ** документе сформировать матрицы $X_{3 \times 5}$ и $Y_{3 \times 3}$ из случайных чисел и вывести их на экран. Для формирования матриц использовать оператор вида

$$X_{i,j} := \text{rnd}(n),$$

где функция $\text{rnd}(n)$ возвращает случайное действительное число из диапазона $(0, n)$, $n = 10$.

а) Используя функцию *augment*, сформировать матрицу Z , содержащую в первых пяти столбцах матрицу X , а в последних трех столбцах – матрицу Y [1, табл. 3.3].

б) Используя функцию *stack*, сформировать матрицу F , содержащую в первых трех строках и в последних трех строках матрицу Y .

в) Вычислить число строк и столбцов матрицы Y , ее наибольший и наименьший элемент, ее ранг и след [1, табл. 3.4].

г) Используя команду M^\wedge , сформировать вектор x как первый столбец матрицы X и вектор y – как первый столбец матрицы Y [1, пример 3.3.1]. Вычислить произведение $\vec{x} \times \vec{y}$.

д) Сформировать диагональную матрицу, используя функцию *diag(x)* [1, табл. 3.2].

Сформировать единичную матрицу размерности 5×5 , используя функцию *identity(n)* [1, табл. 3.2].

Часть III. Работа с файлами в Mathcad

В Mathcad есть возможность чтения данных из файлов и записи данных в файлы.

Чтение данных. Вначале в программе «Блокнот» («Notepad») создается текстовый файл с расширением *.txt. В каждой строке файла с первой позиции через один пробел (или символ табуляции) нужно записать элементы матрицы. Количество строк в документе должно быть равно количеству строк матрицы. Число элементов в каждой строке должно быть одинаковым. Файл, у которого в строках и столбцах записаны числа, называется структурированным. Удобнее сохранить документ в той же папке, где находится документ Mathcad.

Для **чтения из файла** используется функция READPRN("полный путь к имени файла"). Если документ Mathcad и текстовый файл находятся в одной папке, то достаточно указать имя файла. Например, для массива A :

$$A := \text{READPRN}(\text{"matrix.txt"})$$

Запись данных осуществляется функцией WRITEPRN("имя файла"). Каждая строка матрицы становится строкой файла. Например, для массива A :

WRITEPRN("raschet.txt"):=A

Данная функция создает *новый файл* с именем raschet.txt. Если такой файл *уже существует*, то функция *стирает* из указанного файла имеющуюся информацию и записывает новую.

Если имеющаяся в файле информация необходимо сохранить, то используют функцию APPENDPRN("имя файла"), которая *дописывает* матрицу в конец файла:

APPENDPRN("имя файла"):=A

Существующий файл должен иметь столько же столбцов, что и матрица A .

Задание 7

а) Создать с помощью «Блокнота» и сохранить в рабочей папке файл **matrix1.txt**, содержащий матрицу $A_{3 \times 6}$ (числа в матрице задать произвольно).

б) В НОВОМ документе Mathcad прочитать матрицу A из файла **matrix1.txt** и вывести ее на экран.

в) Вычислить матрицу B как транспонированную матрицу A , вывести ее на экран. Записать матрицу B в файл **matrix2.txt**.

г) Сформировать матрицу C как произведение матрицы A на матрицу B , вывести ее на экран. Дописать матрицу C в файл **matrix2.txt**. Открыть файл **matrix2.txt** и проверить его содержимое.

Лабораторная работа № 4 Построение графиков

Для построения графиков в Mathcad существует две возможности:

- 1) использование функций;
- 2) использование массивов (векторов или матриц).

Возможно построение как двумерных, так и трехмерных графиков.

Для построения используются декартова и полярная системы координат.

Задача 1

а) Построить график функции $f(x) = 2x + 3\sin(x)$ в декартовой системе координат [1, § 4.1]. Изменить диапазон аргумента: $x \in [0, 5]$.

б) На одном рисунке построить графики первой и второй производной функции $f(x)$.

Задача 2

а) Сформировать два вектора по 25 элементов каждый по формулам $y_i = \sin(i)/i$, $z_i = \cos(i)$. Построить графики векторов y и z , по оси абсцисс указать индекс i , по оси ординат – индексные переменные y_i и z_i [1, пример 4.2.2].

б) Двойным щелчком на области графика открыть окно форматирования рисунка. Добавить линии сетки (*Grid lines*), изменить цвет, тип и толщину линий.

Задача 3

Дана матрица A с элементами

0	1	3	1
1	4	2	1
2	9	1	1
3	4	0	1

Построить графики столбцов матрицы A . Для этого сформировать четыре вектора, используя операцию выбора столбца (например, $x1 := A^{(1)}$). Задать дискретную переменную (индекс

элемента вектора). Указать имя этой переменной в качестве аргумента на графике. На оси ординат указать имя индексной переменной, например, x_{1i} .

Задача 4

Для параметра w , изменяющегося от 0 до 2π с шагом 0.1 построить графики параметрически заданных функций $x(r, w) = r(w) \cdot \cos(w)$; $y(r, w) = r(w) \cdot \sin(w)$, где функция $r(w)$ определяется выражением $r(w) = 5 \cdot \sin(w) \cdot \cos(w)$ [1, пример 4.2.3].

Задача 5

Для $\varphi = 0, 0.01, \dots, 2\pi$ построить графики $\rho(\varphi) = \cos(2\varphi)$ и $\mu(\varphi) = \sin(2\varphi)$ в полярной системе координат [1, § 4.3].

Задача 6

а) Построить 3D-график функции двух переменных $f(x, y) = x^2 + y^2$, где $x_i = -1.5 + 0.1 \cdot i$; $y_j = -1.5 + 0.1 \cdot j$, $i = 0, 1, \dots, N$; $j = 0, 1, \dots, N$; $N = 30$. Для построения использовать матрицу [1, § 4.4].

б) Изменить вид графика: добавить заливку выбранным цветом, удалить линии каркаса, добавить подсветку. Выполнить вращение графика и изменить масштаб.

в) Для функции из пункта а) построить график изолиний.

Лабораторная работа № 5
Программирование линейных
и разветвляющихся алгоритмов

Часть I. Линейные вычислительные алгоритмы

Задача 1

Написать подпрограммы-функции для решения задач и протестировать их:

а) вычислить площадь треугольника, заданного длинами сторон a, b, c , по формуле Герона;

б) для цилиндра радиусом r и высотой h вычислить площадь и объем по формулам

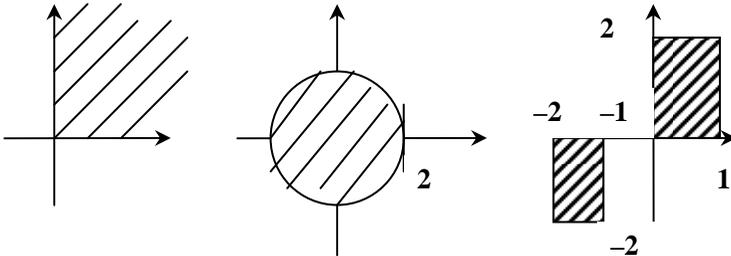
$$s = 2\pi r(h + r); \quad v = \pi r^2 h.$$

Для вывода результатов использовать массив.

Часть II. Разветвляющиеся вычислительные алгоритмы

Задача 2

Используя логические функции, записать выражение, принимающее значение 1 при попадании точки с координатами (x, y) в заштрихованную область и 0 – в противном случае [1, пример 7.2.4].



Задача 3

Построить график функции согласно варианту (табл. 1). Номер варианта соответствует последней цифре номера по журналу. Задачу решить двумя способами:

а) используя условную функцию *if* (безмодульное программирование);

б) используя подпрограмму-функцию (модульное программирование).

Таблица 1

Вариант	Функция	Значения параметров и аргумента
1	$y = \begin{cases} \sin(ax), & \text{если } x < 0 \\ \cos(x/b), & \text{если } 0 \leq x < 1 \\ \sin(x^2), & \text{если } 1 \leq x \end{cases}$	$a = 2, b = 0.5,$ x принимает значения от -5 до 5 с шагом 0.05
2	$y = \begin{cases} \frac{gt^2}{2}, & \text{если } t < a \\ \frac{1}{2}ga^2 + v(t-a), & \text{если } t \geq a \end{cases}$	$a = 2^b, b = 1.75,$ $v = 27.9 \cdot 10^3, g = 9.8,$ t принимает значения от 0 до 5 с шагом 0.02
3	$z = \begin{cases} 2ax + y, & \text{если } x + y < 0.2 \\ 2.4, & \text{если } x + y = 0.2 \\ x^2 \ln(2y^2 + 1), & \text{если } x + y > 0.2 \end{cases}$	$y = 2, a = 1.2,$ x принимает значения от -2 до 5 с шагом 0.02
4	$z = \begin{cases} y^2 + 12x, & \text{если } y \geq 0.1 \\ \sin(2y), & \text{если } y < 0.1 \end{cases}$	$x = \max(a, b),$ $a = \sin(\pi/10),$ $b = \cos(\pi/3),$ y принимает значения от -1 до 2 с шагом 0.02
5	$y = \begin{cases} t, & \text{если } x \leq 0 \\ t + \frac{\sin(x)}{2}, & \text{если } 0 < x < c \\ t + 2, & \text{если } x \geq c \end{cases}$	$t = 2, c = 1.4,$ x принимает значения от -1 до 2 с шагом 0.02
6	$y = \begin{cases} 2x^2 + t^2, & \text{если } x \leq 0 \\ c\sqrt{2}, & \text{если } 0 < x < 2 \\ x + c, & \text{если } x \geq 2 \end{cases}$	$t = 1.47, c = 5.1,$ x принимает значения от -1 до 3 с шагом 0.05
7	$t = \begin{cases} 0.54 \ln(z^2 + 1), & \text{если } z < -0.24 \\ 2.58 + bz, & \text{если } z = -0.24 \\ \cos(2z), & \text{если } z > -0.24 \end{cases}$	$b = 1.47,$ z принимает значения от -1 до 1 с шагом 0.05

Вариант	Функция	Значения параметров и аргумента
8	$y = \begin{cases} x^2 + \operatorname{tg}(x), & \text{если } x \leq 0 \\ b, & \text{если } 0 < x \leq 1 \\ \sqrt{x+1} + \ln x , & \text{если } x > 1 \end{cases}$	$b = 1.47$, x принимает значения от -1 до 3 с шагом 0.05
9	$f = \begin{cases} -2 + \sin(x), & \text{если } x < 1 \\ \sqrt{x}, & \text{если } 1 \leq x < 2 \\ a + x^2, & \text{если } x \geq 2 \end{cases}$	$a = 1.57$, x принимает значения от -1 до 4 с шагом 0.05
10	$z = \begin{cases} -2a, & \text{если } x < -2 \\ x + 1, & \text{если } -2 \leq x < 1 \\ x^2, & \text{если } x \geq 1 \end{cases}$	$a = 1.57$, x принимает значения от -3 до 2 с шагом 0.05

Пример выполнения задачи 3

Построить график функции

$$z = \begin{cases} \sin(x - 2a), & \text{если } x < -2, \\ x + 1, & \text{если } -2 \leq x < 1, \\ x^2, & \text{если } x \geq 1, \end{cases}$$

где $a = 1.47$, x принимает значения от -3 до 2 с шагом 0.05 .

Вариант а), безмодульное программирование

Определим функцию пользователя $z(a, x)$, значения x зададим с помощью дискретного аргумента:

$$x := -3, -2.95..2$$

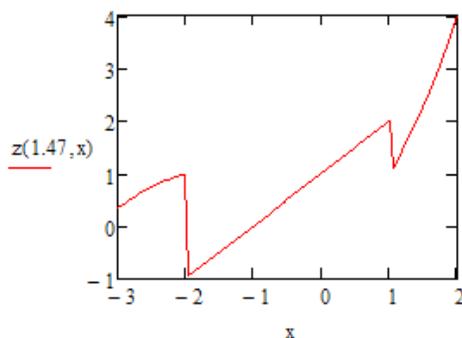
$$z(a, x) := \operatorname{if}(x < -2, \sin(x - 2 \cdot a), \operatorname{if}(-2 \leq x < 1, x + 1, x^2))$$

Вариант б), модульное программирование

Реализуем подпрограмму-функцию для вычисления значения $z(a, x)$ и построим ее график:

$$z(a, x) := \begin{cases} \sin(x - 2 \cdot a) & \text{if } x < -2 \\ x + 1 & \text{if } -2 \leq x < 1 \\ x^2 & \text{otherwise} \end{cases}$$

Построим график с использованием команд *Insert – Graph – X–Y Plot*:



Задача 4

Известны длины сторон треугольника a , b , c . Составить описание функции, результатом работы которой были бы следующие сообщения:

- **Треугольник равносторонний** – если все стороны равны;
- **В треугольнике равны две стороны** – если равны две любые стороны;
- **В треугольнике все стороны не равны.**

Провести тестирование функции.

Задача 5

Написать функцию, которая для точки с координатами (x, y) вычисляет полярные координаты (ρ, φ) , если точка попадает внутрь заштрихованной области из задачи 2. Вид области выбрать самостоятельно. Провести тестирование функции.

Лабораторная работа № 6

Программирование простых циклов

Циклические алгоритмы содержат один или несколько повторяющихся операторов.

Базовыми циклическими алгоритмами являются:

- накопление суммы;
- накопление количества (организация счетчика);
- накопление произведения;
- формирование массивов;
- поиск максимума (минимума).

Для реализации циклов в Mathcad существуют операторы цикла **while** (цикл с предусловием) и **for** (цикл с параметром). Работу этих операторов демонстрирует блок-схема, показанная на рис. 1а. Повторяющиеся операторы образуют **тело цикла**. Тело цикла выполняется до тех пор, пока логическое выражение (ЛВ), входящее в **заголовок цикла**, равно «истина», т.е. 1. Как только ЛВ становится равным «ложь», т.е. 0, происходит выход из цикла, и управление передается на следующий оператор.

Алгоритм накопления суммы. Вычисление суммы начинается с того, что соответствующая переменная **до начала** работы цикла обнуляется: $s \leftarrow 0$. **В теле цикла** накопление суммы организовывается с помощью оператора присваивания вида $s \leftarrow s + x$ (см. рис. 1б).

Алгоритм накопления количества. **До начала** работы цикла переменная для расчета количества обнуляется: $k \leftarrow 0$. **В теле цикла** переменная увеличивается **на единицу**: $k \leftarrow k + 1$.

Алгоритм накопления произведения. **До начала** работы цикла в соответствующую переменную записывается **единица**: $p \leftarrow 1$, а **в теле цикла** накопление произведения организовывается с помощью оператора вида $p \leftarrow p \cdot x$.

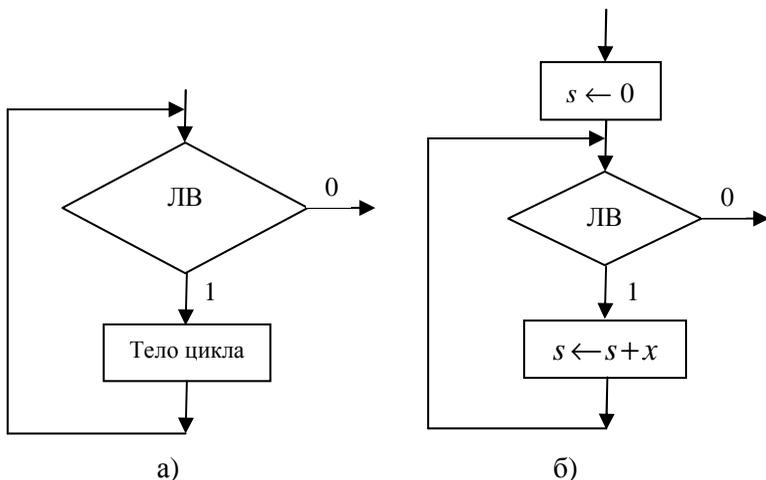


Рис. 1. Блок-схемы циклических алгоритмов

Пример 1

Для последовательности $y(x) = \frac{x+1}{e^x + \sin(x)}$ найти среднее арифметическое всех значений и произведение положительных значений; x меняется от 0 до $\pi/2$ с шагом $\pi/20$.

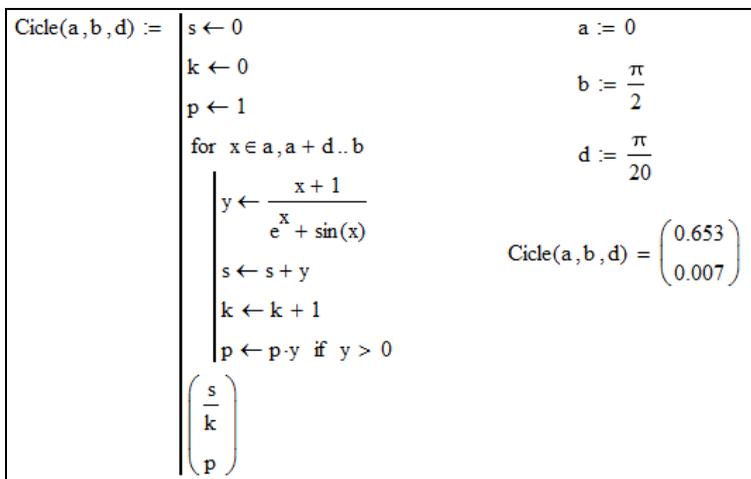


Рис. 2. Пример накопления суммы, количества и произведения

Пояснение: В заголовке цикла *for* для переменной x учтены начальное значение a , следующее значение с учетом шага $a + d$, последнее значение b (рис. 2).

Задача 1

Согласно варианту задания (номер варианта соответствует последней цифре номера в журнале преподавателя, 0 – вариант № 10) написать и протестировать подпрограмму-функцию обработки последовательности.

1. Найти количество положительных и количество отрицательных элементов последовательности $a(k) = \cos(2k - 1)$, $k = 1, 2, \dots, 10$.

2. Найти сумму положительных элементов последовательности и их количество: $\sin(0.1), \sin(0.2), \dots, \sin(10)$.

3. Найти сумму положительных и количество отрицательных элементов последовательности $f(i) = \sin\left(i^2 + \sqrt{|2i - 8|}\right)$, где $i = 1, 2, \dots, 10$.

4. Дана последовательность $d(n) = \sin(n)$, $n = 1, 2, \dots, 50$. Вычислить произведение элементов последовательности, удовлетворяющих условию $|d(n)| > 0.5$, и их количество.

5. Дана последовательность $a(k) = \sin(k^2)e^k$; $k = 1, 2, \dots, 5$. Найти сумму и произведение ненулевых элементов.

6. Вычислить количество и сумму тех элементов последовательности $a(k) = \ln(k^2 - 4k)$; $k = 1, 2, \dots, 8$, которые больше 10.

7. Для последовательности $b(i) = 2i \sin(2i)$, $i = 1, 2, \dots, 10$, найти произведение и сумму ненулевых элементов.

8. Для последовательности $a(k) = \cos(b + 2k)$, $k = 1, 2, \dots, 6$, и $b = 0.1$ найти сумму и количество положительных элементов.

9. Дана последовательность $f(n) = \sin(n + 1)\cos(n + 2)$, $n = 1, 2, \dots, 50$. Вычислить среднее арифметическое положительных элементов этой последовательности.

10. Найти разность между количеством положительных и количеством отрицательных элементов последовательности $a(k) = k^3 3 \cos(k) e^{-k+2}$, $k = 1, 2, \dots, 7$.

Обработка одномерных массивов (векторов)

Пример 2

Написать подпрограмму-функцию, которая в векторе размерностью n вычисляет сумму и количество элементов, принадлежащих интервалу (a, b) . Вызвать подпрограмму-функцию для случайных векторов x_5 и y_7 . Значения a, b задать самостоятельно (рис. 3).

$$\begin{array}{l}
 \text{Vector}(x, n, a, b) := \left\{ \begin{array}{l} s \leftarrow 0 \qquad i := 1..5 \quad x_i := \text{rnd}(10) \\ k \leftarrow 0 \qquad i := 1..7 \quad y_i := \text{rnd}(10) \\ \text{for } i \in 1..n \\ \quad \text{if } a < x_i < b \\ \qquad \left\{ \begin{array}{l} s \leftarrow s + x_i \\ k \leftarrow k + 1 \end{array} \right. \\ \left(\begin{array}{l} "s=" \quad s \\ "k=" \quad k \end{array} \right) \end{array} \right. \\
 \\
 \text{Vector}(x, 5, 0, 2) = \left(\begin{array}{l} "s=" \quad 4.129 \\ "k=" \quad 3 \end{array} \right) \\
 \\
 \text{Vector}(y, 7, 9, 10) = \left(\begin{array}{l} "s=" \quad 9.885 \\ "k=" \quad 1 \end{array} \right)
 \end{array}$$

Рис. 3. Пример обработки одномерного массива (вектора)

Пояснение: Для округления действительного числа x до целого можно использовать функции $\text{round}(x)$, $\text{trunc}(x)$, $\text{ceil}(x)$, $\text{floor}(x)$, для формирования вектора x размерности n из случайных чисел – функции $\text{rnd}(n)$, $\text{runif}(n, a, b)$, $\text{rnorm}(n, mx, dx)$ (см. лабораторную работу № 3).

Задача 2

Согласно варианту задания написать подпрограмму-функцию обработки одномерного массива (вектора) и протестировать ее.

1. В целочисленном массиве размерностью n вычислить сумму положительных элементов и количество четных (по значению) элементов массива. Вызвать подпрограмму-функцию для случайных векторов x_5 и y_7 .

2. Элементы массива вычисляются по формуле $y_i = 1/(i + 2)$, $i = 1, 2, \dots, 15$. Найти отношение суммы четных (по номеру) элементов массива к сумме нечетных (по номеру) элементов.

3. Дан массив размерностью n ($n > 6$), содержащий положительные, отрицательные и нулевые элементы. Все отрицательные элементы массива заменить произведением его ненулевых элементов.

4. Элементы массива вычисляются по формуле $y_i = 1/(i + 2)$, $i = 1, 2, \dots, 15$. Найти произведение суммы элементов массива, меньших 0.1, и сумму элементов, меньших 0.5.

5. Элементы массива y ($n = 7$) формируются как случайные целые числа из диапазона $[-5, 5]$. Найти произведение ненулевых элементов, удовлетворяющих условию $a \leq |y_i| \leq b$, a и b задать самостоятельно.

6. Дан массив случайных чисел размерностью n . Заменить все элементы массива с четными индексами средним арифметическим всех элементов массива.

7. Дан массив случайных чисел размерностью n . Найти произведение положительных элементов с нечетными индексами.

8. Элементы массива x ($n = 7$) формируются как случайные целые числа из диапазона $[0, 10]$. Вычислить среднее арифметическое элементов вектора, а затем преобразовать исходный вектор по правилу: те элементы, которые меньше среднего арифметического, заменить нулями.

9. Даны два вектора x и y , состоящие из n элементов. В подпрограмме-функции сформировать вектор q по правилу: q_i -й элемент равен **1**, если точка с координатами (x_i, y_i) принадлежит кругу радиусом r с центром в начале координат, и **0** в противном случае.

10. Даны два массива произвольной размерности, элементы которых вычисляются как $\lambda_i = 1/(i + \sin(i^2))$, $\mu_i = \cos(i + 5)$. Найти количество положительных элементов в этих массивах. Вычисление количества в одном векторе оформить функцией.

Алгоритм поиска максимума (минимума) в последовательности (векторе). До цикла в переменную для расчета максимума записывается большое отрицательное число, в переменную для расчета минимума записывается большое положительное число: $\max \leftarrow -10^6$, $\min \leftarrow 10^6$. Можно также записать первый элемент вектора: $\max \leftarrow x_1$, $\min \leftarrow x_1$. В теле цикла последовательно сравниваются максимум (минимум) с элементами вектора, и при выполнении условия в \max (\min) записывается новое значение: $\max \leftarrow x_i$ if $x_i > \max$, $\min \leftarrow x_i$ if $x_i < \min$.

Пример 3

Из случайных целых чисел из диапазона [0, 10] сформировать одномерный массив (вектор) x , состоящий из n элементов. Составить подпрограмму-функцию для поиска максимального элемента этого массива и его номера (индекса).

```

ORIGIN := 1
Max_number(x) :=
  max ← x1
  imax ← 1
  for i ∈ 2..length(x)
    if xi > max
      max ← xi
      imax ← i
  ( "max" max
    "number" imax )
  i := 1..5
  xi := round(rnd(10))
  x = ( 10
        1
        0
        5
        6 )
  Max_number(x) = ( "max" 10
                   "number" 1 )

```

Рис. 4. Пример нахождения максимума в векторе и его номера

Пояснение: В примере (рис. 4) цикл *for* начинается со значения **2**, так как **1** уже была учтена в операторах в начале программы. Для определения количества элементов в векторе использована функция *length(x)*. Тогда передавать размерность вектора как параметр функции не нужно.

Задача 3

Согласно варианту задания написать подпрограмму-функцию обработки одномерных массивов (векторов) и протестировать ее.

1. Элементы массива y ($n = 7$) формируются как случайные целые числа из диапазона $[0, 5]$. Вычислить разность между максимальным и минимальным элементами массива.

2. Из случайных целых чисел в диапазоне $[-5, 5]$ сформировать одномерный массив (вектор) x , состоящий из n элементов. Заменить минимальный элемент массива средним арифметическим всех его элементов.

3. Элементы массива вычисляются по формуле: $y_i = 2^i \cos(2i) / \sin(2 + i)$, $i = 1, 2, \dots, 10$. Найти минимальный по модулю элемент этого массива и его порядковый номер.

4. Дан массив $x^T = (28.1, -5, 2.6, 1.3, 0, -1.1, 2.1, 13.0)$. Поменять местами его максимальный и нулевой элементы.

5. Дан массив $x^T = (4, 0, 32, 10, 9.2, 2.13, 38, 5.2)$. Найти отношение номера минимального элемента массива к номеру максимального элемента.

6. Элементы массива вычисляются по формуле $a_i = \cos(i/10)$, где $i = 1, 2, \dots, n$; $n = 10$. Найти максимальный и минимальный элементы массива и их порядковые номера.

7. Даны два вектора x , y , состоящие из n элементов каждый. Вычислить полусумму их максимальных значений. Вычисление максимальной величины вектора оформить функцией.

8. Даны два массива произвольной размерности, элементы которых вычисляются как $\lambda_i = 1 / (i + \sin(i^2))$, $\mu_i = \cos(i + 5)$. Найти максимальные и минимальные элементы в этих массивах.

Вычисление максимальной и минимальной величин одного вектора оформить функцией.

9. Дан массив $b^T = (6, -7, 2, 4, -2, 3)$. Среди элементов с четными индексами найти минимальный элемент.

10. Дан массив $x^T = (28.1, -5, 2.6, 1.3, 0, -1.1, 2.1, 13.0)$. Среди элементов, имеющих нечетный индекс, найти максимальный элемент.

Итерационные циклические алгоритмы

Если в алгоритме количество повторений цикла заранее неизвестно, то такие алгоритмы называются *итерационными*. Каждое повторение операторов, находящихся в теле цикла, называется *итерацией*. Количество итераций зависит от заданной *точности*: чем выше точность, тем больше количество итераций.

Для организации итерационных циклов (рис. 5) используется оператор цикла с предусловием: *while <логическое выражение>*, который выполняется до тех пор, пока логическое выражение истинно (равно **1**).

Пример 4

Вычислить приближенное значение \sqrt{a} , используя итерационную процедуру:

$$x_{n+1} = 0.5 \cdot \left(x_n + \frac{a}{x_n} \right), \quad n = 0, 1, \dots; \quad x_0 = a.$$

В качестве приближенного значения \sqrt{a} принимается x_{n+1} , удовлетворяющее условию $|x_{n+1} - \sqrt{a}| \leq \varepsilon$, где ε – заданная точность вычисления корня квадратного, n – число итераций.

Задача 4

Согласно варианту задания написать и протестировать подпрограмму-функцию при разной точности ε .

1. Найти сумму сходящегося ряда $\sum_{n=1}^{\infty} (-1)^{n+1} \frac{\sin(nx)}{n}$ при $x = 0.75$. Суммирование производить до тех пор, пока слагаемые больше ε .

2. Найти сумму сходящегося ряда $\cos(x) + \frac{\cos(3x)}{2} + \frac{\cos(5x)}{3} + \dots$ при $x = 0.5$. Суммирование производить до тех пор, пока слагаемые больше ε .

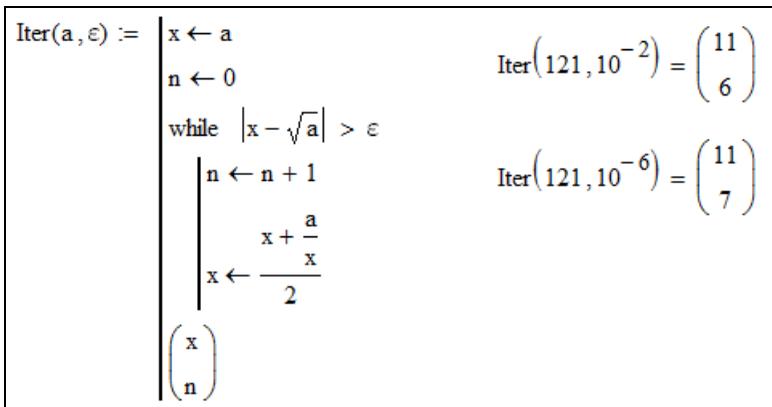


Рис. 5. Пример итерационного алгоритма

3. Написать подпрограмму-функцию численной проверки первого замечательного предела $\lim_{x \rightarrow 0} \left(\frac{\sin(x)}{x} \right) = 1$. Задаются значения $x = 1, 1/2, 1/4, 1/8$ и т.д. до тех пор, пока левая часть равенства не будет отличаться от правой менее чем на заданную погрешность ε .

4. Написать подпрограмму-функцию численной проверки второго замечательного предела $\lim_{n \rightarrow \infty} \left(1 + \frac{1}{n} \right)^n = e$. Задаются

значения $n = 1, 2, 3$ и т.д. до тех пор, пока левая часть равенства не будет отличаться от правой менее чем на заданную погрешность ε .

5. Написать подпрограмму-функцию разложения в ряд числа π с заданной точностью ε по формуле

$$\pi = \sqrt{6 \left(1 + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \dots \right)}.$$

6. Написать подпрограмму-функцию, которая вычисляет, сколько сомножителей надо взять в произведении

$$\prod_{k=1}^{\infty} \left(1 + \frac{(-1)^k}{2k+1} \right) = \frac{\sqrt{2}}{2},$$

чтобы равенство выполнялось с точностью ε .

7. Написать подпрограмму-функцию, которая вычисляет, сколько сомножителей надо взять в произведении

$$\prod_{k=2}^{\infty} \left(1 - \frac{1}{k^2} \right) = \frac{1}{2},$$

чтобы равенство выполнялось с точностью ε .

8. Написать подпрограмму-функцию, которая для заданных значений a и p вычисляет $x = \sqrt[p]{a}$, используя формулу

$$x_{n+1} = \frac{1}{p} \left[(p-1)x_n + \frac{a}{x_n^{p-1}} \right]; \quad x_0 = a.$$

Вычисления выполнять до тех пор, пока $|x_{n+1} - \sqrt[p]{a}| > \varepsilon$.

9. Написать подпрограмму-функцию разложения в ряд $\operatorname{arctg}(x)$ с заданной точностью ε по формуле

$$\operatorname{arctg}(x) = x - \frac{x^3}{3} + \frac{x^5}{5} - \dots + (-1)^n \frac{x^{2n+1}}{2n+1} + \dots$$

Значение $|x| < 1$ задать самостоятельно.

10. Написать подпрограмму-функцию разложения в ряд $\ln(x)$ с заданной точностью ε по формуле

$$\ln(x) = 2 \left[\frac{x-1}{x+1} + \frac{1}{3} \left(\frac{x-1}{x+1} \right)^3 + \dots + \frac{1}{2n-1} \left(\frac{x-1}{x+1} \right)^{2n-1} \right].$$

Значение $x > 0$ задать самостоятельно.

Лабораторная работа № 7

Программирование двойных циклов

Для обработки двумерных массивов (матриц) используются **двойные циклы**. Двойные циклы – это конструкции, состоящие из двух циклов – внешнего и внутреннего. Для их построения можно использовать любые сочетания операторов *for* и *while*. При этом важно соблюдать следующие правила:

- переменные, управляющие внешним и внутренним циклами, должны иметь разные имена;
- внутренний цикл заканчивается быстрее внешнего;
- переменные, управляющие циклами, не должны менять своих значений в теле циклов.

Пример 1

Написать подпрограмму-функцию вычисления суммы и количества положительных элементов матрицы. Протестировать работу функции на случайных целочисленных матрицах $A_{3 \times 4}$ и $B_{4 \times 2}$ (рис. 6).

<pre> ORIGIN := 1 Matrix(A ,n ,m) := s ← 0 k ← 0 for i ∈ 1..n for j ∈ 1..m if A_{i,j} > 0 s ← s + A_{i,j} k ← k + 1 ("summa =" s) ("kolichestvo =" k) Matrix(A ,3 ,4) = ("summa =" 18) ("kolichestvo =" 6) Matrix(B ,4 ,2) = ("summa =" 10) ("kolichestvo =" 4) </pre>	<pre> i := 1..3 j := 1..4 A_{i,j} := round(rnd(10) - 5) A = (1 -5 -4 4) (2 -1 1 5) (-3 -4 5 -5) i := 1..4 j := 1..2 B_{i,j} := round(rnd(10) - 5) B = (-4 1) (0 -2) (3 -2) (4 2) </pre>
--	---

Рис. 6. Пример алгоритма типа I для обработки матриц

Пояснение: Все задачи обработки матриц можно свести к двум типам. В задачах **I типа** производится обработка **всех** элементов матрицы. В задачах **II типа** обработка элементов матрицы происходит **по строкам** или **по столбцам**. Задача из примера 1 относится к задачам I типа.

Задачи 1, 2

Согласно варианту задания написать подпрограммы-функции обработки двумерных массивов (матриц) и протестировать их.

1.1. Составить функцию, формирующую матрицу A по следующему правилу:

$$A_{i,j} = \begin{cases} \sin(i + j), & \text{если } i = j; \\ \sin(i) + \cos(j), & \text{если } i > j; \\ \sin(j) + \cos(i), & \text{если } i < j. \end{cases}$$

Сформировать с ее помощью матрицы $X_{5 \times 5}$ и $Y_{3 \times 4}$.

1.2. Дана случайная матрица X размерностью $n \times m$. Составить функцию, вычисляющую среднее арифметическое элементов матрицы, а затем преобразующую исходную матрицу по правилу: те элементы, которые меньше среднего арифметического, заменить нулями.

2.1. Написать подпрограмму-функцию, которая преобразует матрицу X , разделив все ее элементы на максимальный элемент. Протестировать работу функции на случайных целочисленных матрицах $X_{4 \times 4}$ и $Y_{3 \times 3}$.

2.2. Даны две случайные матрицы A, B размерностью $n \times m$. Составить подпрограмму-функцию, подсчитывающую число случаев $A_{i,j} \geq B_{i,j}$. Матрицы A, B задать самостоятельно.

3.1. Дана произвольная квадратная матрица C размерности $n \times n$. Составить подпрограмму-функцию, вычисляющую число элементов, находящихся ниже главной диагонали и удовлетворяющих условию $a \leq C_{i,j} \leq b$. Протестировать работу функции на отрезках $[0, 5]$, $[-2, 2]$.

3.2. Дана произвольная матрица A размерности $n \times m$. Составить подпрограмму-функцию, вычисляющую сумму и произведение только тех элементов матрицы, которые по модулю больше некоторой величины ε . Протестировать работу функции для разных значений ε .

4.1. Даны две матрицы разной размерности $A_{n \times m}$ и $B_{k \times g}$. Вычислить величину $\max A + \max B$, где $\max A$ и $\max B$ – это максимальные элементы матриц A , B соответственно. Вычисление максимального элемента матрицы оформить подпрограммой-функцией.

4.2. Написать подпрограмму-функцию, которая в заданной матрице заменяет все положительные элементы нулями, если количество отрицательных элементов окажется больше, чем количество положительных элементов. Протестировать работу функции на случайных целочисленных матрицах $P_{4 \times 4}$ и $Q_{3 \times 3}$.

5.1. Написать подпрограмму-функцию вычисления евклидовой нормы матрицы $A_{n \times m}$ по формуле:

$$\|A\|_E = \sqrt{\sum_{i=1}^n \sum_{j=1}^m a_{i,j}^2}. \text{ Протестировать работу функции на случай-}$$

ных матрицах $A_{4 \times 4}$ и $B_{3 \times 3}$.

5.2. Дана целочисленная случайная матрица X размерности $n \times n$. Составить подпрограмму-функцию, вычисляющую число положительных элементов, находящихся ниже главной диагонали, и их сумму. Протестировать работу функции на случайных целочисленных матрицах $X_{4 \times 4}$, $Y_{3 \times 3}$, $Z_{5 \times 5}$.

6.1. Даны две матрицы разной размерности $A_{n \times m}$ и $B_{k \times g}$. Вычислить величину $P1 + P2$, где $P1$ и $P2$ – произведения ненулевых элементов матриц A и B соответственно. Для вычисления произведения ненулевых элементов составить подпрограмму-функцию.

6.2. Дана целочисленная случайная матрица X размерности $n \times n$. Составить подпрограмму-функцию, вычисляющую мак-

симальный и минимальный элементы матрицы. Протестировать работу функции на случайных целочисленных матрицах $A_{4 \times 4}$ и $B_{3 \times 3}$.

7.1. Дана целочисленная случайная матрица X размерности $n \times n$. Составить подпрограмму-функцию, вычисляющую сумму и произведение элементов, находящихся выше главной диагонали. Протестировать работу функции на случайных целочисленных матрицах $A_{4 \times 4}$ и $B_{3 \times 3}$.

7.2. Для заданной матрицы составить подпрограмму-функцию вычисления и печати квадрата суммы положительных элементов и суммы квадратов отрицательных элементов матрицы. Протестировать работу функции на случайных целочисленных матрицах X и Y произвольной размерности.

8.1. Дана целочисленная случайная матрица A размерности $n \times n$. Составить подпрограмму-функцию, вычисляющую число положительных элементов, находящихся ниже главной диагонали, и число положительных элементов, находящихся выше главной диагонали. Протестировать работу функции на случайных целочисленных матрицах $A_{4 \times 4}$ и $B_{3 \times 3}$.

8.2. Для заданной матрицы написать подпрограмму-функцию вычисления величины $L = \max - |\min|$, где \max – целая часть максимального элемента, \min – целая часть минимального элемента матрицы A . Протестировать работу функции на случайных целочисленных матрицах X и Y произвольной размерности.

9.1. Дана целочисленная случайная матрица A размерности $n \times n$. Составить подпрограмму-функцию, вычисляющую след матрицы (след матрицы – это сумма элементов, лежащих на главной диагонали). Протестировать работу функции на случайных целочисленных матрицах $A_{4 \times 4}$ и $B_{3 \times 3}$.

9.2. Дана целочисленная случайная матрица X размерности $n \times n$. Составить подпрограмму-функцию, вычисляющую минимальный элемент матрицы и номера его строки и столбца. Протестировать работу функции на случайных целочисленных матрицах $A_{4 \times 4}$ и $B_{3 \times 3}$.

10.1. Даны две матрицы разной размерности $A_{n \times m}$ и $B_{k \times g}$. Вычислить величину $S1 - S2$, где $S1$ и $S2$ – суммы положительных элементов матриц A и B соответственно. Для вычисления суммы составить подпрограмму-функцию.

10.2. Написать подпрограмму-функцию, которая в заданной матрице заменяет элементы, расположенные ниже главной диагонали, нулями. Протестировать работу функции на случайных целочисленных матрицах X и Y произвольной размерности.

Пример 2

Написать подпрограмму-функцию, формирующую вектор, i -й элемент которого равен количеству положительных элементов в i -й строке матрицы. Протестировать работу функции на целочисленных квадратных матрицах $X_{4 \times 4}$ и $Y_{3 \times 3}$.

В примере (рис. 7) используются стандартные функции $rows(A)$, $cols(A)$ для определения числа строк и числа столбцов в обрабатываемой матрице. Их применение позволяет не указывать размерность матрицы как параметры функции, достаточно указать только имя матрицы.

<pre> ORIGIN := 1 Matrix2(A) := for i ∈ 1..rows(A) kp ← 0 ko ← 0 for j ∈ 1..cols(A) kp ← kp + 1 if A_{i,j} > 0 ko ← ko + 1 if A_{i,j} < 0 v_i ← kp - ko v </pre>	$X := \begin{pmatrix} -1 & 2 & -2 & -2 \\ 1 & 2 & 3 & 4 \\ -3 & -1 & 2 & 4 \\ 0 & 0 & 1 & 0 \end{pmatrix}$	$\text{Matrix2}(X) = \begin{pmatrix} 2 \\ 4 \\ 0 \\ 1 \end{pmatrix}$
	$Y := \begin{pmatrix} 0 & 5 & 0 \\ -1 & -2 & -3 \\ 2 & 3 & -7 \end{pmatrix}$	$\text{Matrix2}(Y) = \begin{pmatrix} 1 \\ 3 \\ 1 \end{pmatrix}$

Рис. 7. Пример алгоритма типа II (по строкам)

Пример 3

Написать подпрограмму-функцию, которая вместо первого элемента j -го столбца записывает сумму всех элементов этого

столбца матрицы. Протестировать работу на матрицах $A_{3 \times 4}$ и $B_{4 \times 3}$.

<pre> ORIGIN := 0 Matrix3(A) := for j ∈ 0..cols(A) - 1 s ← 0 for i ∈ 0..rows(A) - 1 s ← s + A_{i,j} A_{0,j} ← s A </pre>	$A := \begin{pmatrix} -1 & 2 & 2 & -2 \\ 1 & 2 & 3 & 4 \\ 0 & 0 & 1 & 0 \end{pmatrix}$	$\text{Matrix3}(A) = \begin{pmatrix} 0 & 4 & 6 & 2 \\ 1 & 2 & 3 & 4 \\ 0 & 0 & 1 & 0 \end{pmatrix}$
<pre> A </pre>	$B := \begin{pmatrix} 0 & 5 & 0 \\ -1 & 2 & -3 \\ 2 & 3 & -7 \\ 0 & 1 & 1 \end{pmatrix}$	$\text{Matrix3}(B) = \begin{pmatrix} 1 & 11 & -9 \\ -1 & 2 & -3 \\ 2 & 3 & -7 \\ 0 & 1 & 1 \end{pmatrix}$

Рис. 8. Пример алгоритма типа II (по столбцам)

Пояснение: В примере (рис. 8) начальное значение индексов матрицы установлено равным *нулю* (ORIGIN:=0). В этом случае циклы начинаются с *нуля* и заканчиваются $rows(A) - 1$, $cols(A) - 1$.

Задачи 3, 4

Согласно варианту задания написать подпрограммы-функции обработки матриц и протестировать их.

1.3. Дана произвольная матрица D размерности $n \times m$. Составить подпрограмму-функцию, которая вместо последнего элемента i -й строки записывает количество элементов i -й строки матрицы D , удовлетворяющих условию $D_{i,j} \geq \delta$, где δ – задаваемая величина.

1.4. Имеется экзаменационная ведомость студенческой группы из $n = 20$ человек по $m = 5$ дисциплинам. Оценки из этой ведомости занесены в матрицу размерности $n \times m$. Составить подпрограмму-функцию, вычисляющую средний балл по каждой дисциплине.

2.3. Дана произвольная матрица A размерности $n \times m$. Составить подпрограмму-функцию, формирующую вектор, i -й элемент которого равен среднему арифметическому i -й строки матрицы.

2.4. Дана случайная матрица X размерности $n \times m$. Составить подпрограмму-функцию, вычисляющую в каждом столбце количество положительных элементов матрицы.

3.3. Имеется экзаменационная ведомость студенческой группы из $n = 20$ человек по $m = 5$ дисциплинам. Оценки из этой ведомости занесены в матрицу размерности $n \times m$. Составить подпрограмму-функцию, вычисляющую средний балл по каждому студенту.

3.4. Дана произвольная матрица A размерности $n \times m$. Составить подпрограмму-функцию, вычисляющую в каждом столбце сумму только тех элементов матрицы, которые удовлетворяют условию $A_{i,j} \geq \delta$, где δ – задаваемая величина.

4.3. Написать подпрограмму-функцию вычисления суммы максимальных элементов строк заданной матрицы.

4.4. Имеется экзаменационная ведомость студенческой группы из $n = 20$ человек по $m = 5$ дисциплинам. Оценки из этой ведомости занесены в матрицу размерности $n \times m$. Составить подпрограмму-функцию, вычисляющую номер дисциплины (номер столбца), по которой у студентов максимальный средний балл.

5.3. Написать подпрограмму-функцию вычисления суммы минимальных элементов строк заданной матрицы Z размерности $n \times m$.

5.4. Написать подпрограмму-функцию, которая в заданной матрице A в каждом столбце находит число положительных элементов и число отрицательных и выводит их в виде матрицы.

6.3. Написать подпрограмму-функцию вычисления произведения сумм элементов каждой строки матрицы A размерности $n \times m$.

6.4. Дана матрица X размерности $n \times m$. Составить подпрограмму-функцию, которая в заданной матрице каждый последний элемент столбца заменяет минимальным элементом этого столбца.

7.3. Дана матрица Y размерности $n \times m$. Составить подпрограмму-функцию вычисления L -нормы заданной матрицы.

Пояснение: L -нормой называется число, равное максимальной из сумм модулей элементов матрицы, расположенных в ее строках.

7.4. Составить подпрограмму-функцию, которая в заданной матрице A каждый первый элемент столбца заменяет произведением элементов этого столбца.

8.3. Написать подпрограмму-функцию умножения матрицы на вектор. Программа должна проверять корректность вводимых данных (размерности матрицы и вектора).

8.4. Написать подпрограмму-функцию, которая в заданной матрице A в каждом столбце находит сумму положительных и сумму отрицательных элементов и выводит их в виде матрицы.

9.3. Написать подпрограмму-функцию, которая в заданной матрице A размерности $n \times m$ каждый последний элемент строки заменяет максимальным элементом этой строки.

9.4. Написать подпрограмму-функцию, которая по заданной матрице Z размерностью $n \times m$ формирует вектор s размерностью m , состоящий из арифметических средних столбцов матрицы Z .

10.3. Написать подпрограмму-функцию, которая по заданной матрице A размерностью $n \times m$ формирует вектор z размерностью n , состоящий из максимальных элементов строк матрицы A .

10.4. Написать подпрограмму-функцию, которая в заданной матрице каждый последний элемент столбца заменяет суммой элементов этого столбца.

Лабораторная работа № 8

Решение нелинейных алгебраических уравнений

Постановка задачи. Решить нелинейное алгебраическое уравнение (НАУ) вида $f(x) = 0$. **Решением уравнения** является значение x^* такое, что $f(x^*) = 0$.

Задача численного решения НАУ состоит из двух этапов. На первом этапе устанавливается, имеет ли уравнение корни, сколько их, выполняется процедура **отделения корней**. На втором этапе **корни уточняются**.

Отделить корень – значит найти такой **интервал изоляции** $[a, b]$, который содержит **один** корень уравнения. Необходимым условием того, что на отрезке $[a, b]$ существует хотя бы один корень уравнения, является условие непрерывности функции на этом отрезке и смена знака функции на концах отрезка, т.е. $f(a) \cdot f(b) < 0$. Корень будет единственным, если $f(a) \cdot f(b) < 0$ и производная $f'(x)$ не меняет знак на отрезке $[a, b]$, т.е. $f(x)$ – монотонная функция. В Mathcad для отделения корней будем использовать графический метод.

Пример

Для уравнения $f(x) = x^3 + 3x^2 - 15x - 12$ определить интервалы изоляции корней.

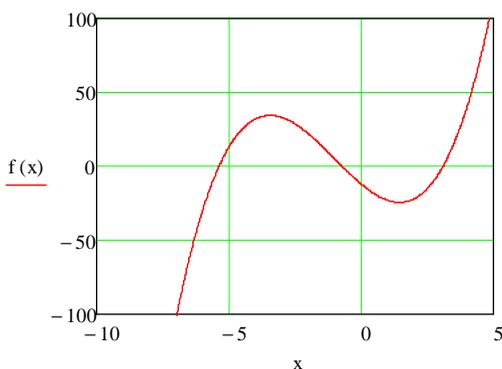


Рис. 9. Графический метод отделения корней

На рис. 9 показан график уравнения. Видно наличие трех корней, для каждого из которых нужно определить свой интервал изоляции. Для первого корня предлагается выбрать интервал $[-10, -5]$, для второго – $[-2, 0]$, для третьего – $[2, 5]$.

Уточнение корней – это вычисление корней с заданной точностью на интервале изоляции.

Использование функций Mathcad для уточнения корней

1) **Функция *roots*()**. В Mathcad для вычисления корней любого нелинейного уравнения используется функция *root*(), которая может иметь два или четыре аргумента, т.е. $root(f(x), x)$ или $root(f(x), x, a, b)$, где $f(x)$ – имя функции или запись непосредственно уравнения; x – переменная, относительно которой решается уравнение; a, b – границы интервала изоляции корня.

$$root(f(x), x, -10, -5) = -5.375$$

$$x := -10 \quad x1 := root(f(x), x) = -5.375$$

$$\text{Проверка: } f(x1) = -7.105 \cdot 10^{-14}$$

2) **Функция *polyroots*()**. Для вычисления всех корней полинома степени n (не выше 5) используют функцию *polyroots*(v), где v – вектор, состоящий из коэффициентов полинома: $v_0 = a_0$, $v_1 = a_1, \dots, v_n = a_n$. Эта функция не требует процедуры отделения корней.

$$v := \begin{pmatrix} -12 \\ -15 \\ 3 \\ 1 \end{pmatrix} \quad x := polyroots(v) = \begin{pmatrix} -5.375 \\ -0.721 \\ 3.096 \end{pmatrix}$$

$$\text{Проверка: } (x_0)^3 + 3 \cdot (x_0)^2 - 15 \cdot x_0 - 12 = 0$$

3) **Блок *Given...Find*()**. Для вычисления корня нелинейного уравнения можно использовать вычислительный блок **Given...Find**. Служебные слова *Given* и *Find* пишутся с прописной

буквы. Задание начальной точки необходимо. В примере поиск решения происходит от начальной точки -10 .

$$f(x) := x^3 + 3 \cdot x^2 - 15 \cdot x - 12$$

$$x := -10$$

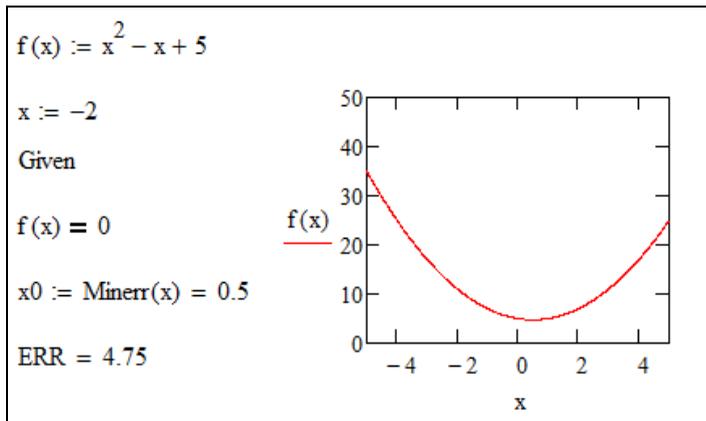
Given

$$f(x) = 0$$

$$\text{Find}(x) = -5.375$$

В записи уравнений используется логический оператор равенства, вводимый с палитры **Boolean**.

Если на заданном интервале $[a, b]$ **корня уравнения не существует**, то следует использовать блок **Given...Minerr**. Функция $\text{Minerr}(x)$ возвращает приближенное значение корня, при котором левая и правая части уравнения наиболее близки. Системная переменная ERR показывает ошибку найденного приближенного решения:



Приближенные методы решения НАО

Решить уравнение приближенным (итерационным) методом – это значит построить последовательность $\{x_n\}$, сходящуюся к точному решению x^* . Здесь n – номер итерации (приближения) к решению. В итерационных методах каждое последующее приближение вычисляется по известному предыдущему. Условиями прекращения вычислений являются:

$$|x^* - x_n| < \varepsilon, \quad (1)$$

$$|f(x_n)| < \varepsilon, \quad (2)$$

$$|x_{n+1} - x_n| < \varepsilon. \quad (3)$$

Здесь ε – точность метода, наперед заданное малое число.

К итерационным методам относятся: метод деления отрезка пополам (дихотомии), метод Ньютона (метод касательных), метод простых итераций.

Метод деления отрезка пополам. Пусть на отрезке $[a, b]$ имеется один корень уравнения. Разделим отрезок $[a, b]$ пополам точкой $c = (a + b)/2$ и проверим, на каком отрезке – $[a, c]$ или $[c, b]$ – функция меняет знак: $f(a) \cdot f(c) < 0$ либо $f(c) \cdot f(b) < 0$. Выберем тот из отрезков, на котором функция меняет знак. Концы нового отрезка обозначим a, b :

$$a_i = \begin{cases} c_{i-1}, & \text{если } f(a_{i-1}) \cdot f(c_{i-1}) > 0; \\ a_{i-1}, & \text{в противном случае} \end{cases}$$
$$b_i = \begin{cases} b_{i-1}, & \text{если } f(a_{i-1}) \cdot f(c_{i-1}) > 0; \\ c_{i-1}, & \text{в противном случае} \end{cases} \quad i = 1, 2, \dots$$

Будем повторять процесс деления до тех пор, пока не будет достигнуто условие сходимости итерационного процесса (2) или $|b - a| < \varepsilon$. На рис. 10 приведено решение НАО методом дихотомии.

$\text{Dichotomy}(a, b, f, \varepsilon) :=$	<pre> return ∞ if $f(a) \cdot f(b) > 0$ while $b - a \geq \varepsilon$ $c \leftarrow \frac{a + b}{2}$ break if $f(c) < \varepsilon$ $a \leftarrow c$ if $f(a) \cdot f(c) > 0$ $b \leftarrow c$ otherwise $\frac{a + b}{2}$ </pre>
$\underline{x} := \text{Dichotomy}(-10, -5, f, 10^{-3}) \quad x = -5.375$	
$f(x) = 7.359 \times 10^{-3}$	

Рис. 10. Пример решения НАУ методом дихотомии

Метод Ньютона (касательных). Выберем начальное приближение $x_0 = a$ или $x_0 = b$ так, чтобы выполнялось условие $f(x_0) \cdot f''(x_0) > 0$. Следующие итерации вычисляются по формуле

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}.$$

Условие окончания итерационного процесса: $|x_{n+1} - x_n| < \varepsilon$.
 На рис. 11 приведено решение НАУ методом Ньютона.

$$\begin{array}{l}
 \text{Newton}(x_0, f, fl, \varepsilon) := \left\{ \begin{array}{l}
 n \leftarrow 0 \\
 \text{while } n \leq 10^5 \\
 \quad \left\{ \begin{array}{l}
 x_1 \leftarrow x_0 - \frac{f(x_0)}{fl(x_0)} \\
 \text{break if } |x_1 - x_0| < \varepsilon \\
 n \leftarrow n + 1 \\
 x_0 \leftarrow x_1
 \end{array} \right. \\
 \end{array} \right. \quad fl(x) := \frac{d}{dx} f(x) \\
 \\
 \underline{x} := -10 \quad f(x) \cdot \frac{d^2}{dx^2} f(x) > 0 = 1 \\
 \text{Newton}(-10, f, fl, 10^{-3}) = -5.375
 \end{array}$$

Рис. 11. Пример решения НАУ методом Ньютона

Метод простых итераций. Приведем исходное уравнение к виду $x = \varphi(x)$. Выбрав нулевое приближение x_0 , подставим его в правую часть и получим следующее приближение к корню $x_1 = \varphi(x_0)$. Будем продолжать итерации до тех пор, пока не будет выполнено условие (2) или (3). Данная итерационная процедура сходится, если выполняется достаточное условие сходимости $|\varphi'(x) < 1|$ для всех $x \in [a, b]$. В противном случае итерационный процесс может расходиться.

Приведем исходное уравнение к виду $x = x - c \cdot f(x)$. Тогда $\varphi(x) = x - c \cdot f(x)$. Параметр c подберем таким образом, чтобы выполнялось достаточное условие сходимости $c = \frac{2}{M + m}$, где

$$M = \max(f'(x)), \quad m = \min(f'(x)) \quad \text{для } x \in [a, b].$$

В качестве стартового значения можно выбрать середину отрезка $[a, b]$: $x_0 = (a + b)/2$. Следующие итерации находим по

формуле $x_{n+1} = x_n - c \cdot f(x_n)$. При выполнении условия (2) или (3) итерации прекращаются. На рис. 12 приведено решение НАУ методом простых итераций.

```

SimpleIteration( $\varphi, x_0, \epsilon$ ) :=  $\left\{ \begin{array}{l} n \leftarrow 0 \\ \text{while } n \leq 10^5 \\ \quad \left\{ \begin{array}{l} x1 \leftarrow \varphi(x0) \\ \text{break if } |x1 - x0| < \epsilon \\ n \leftarrow n + 1 \\ x0 \leftarrow x1 \end{array} \right. \\ x1 \end{array} \right.$ 

a := -10    b := -5

 $\underline{\underline{f1}}(x) := \frac{d}{dx} f(x)$     M := max(f1(a), f1(b))     $\underline{\underline{m}} := \min(f1(a), f1(b))$ 

 $\underline{\underline{c}} := \frac{2}{M + m} = 0.008$      $\varphi(x) := x - c \cdot f(x)$ 

 $\underline{\underline{x}} := \text{SimpleIteration}\left(\varphi, \frac{a + b}{2}, 10^{-3}\right) = -5.377$     f(x) = -0.065

```

Рис. 12. Пример решения НАУ методом простых итераций

Задание

1. Исследовать функцию $f(x)$ на наличие корней графически. Найти и выписать интервалы, на которых существует единственный корень уравнения.

2. Для каждого интервала изоляции $[a, b]$ найти корни уравнения с использованием функций Mathcad $root()$, $polyroots()$, блока *Given...Find*.

3. Для каждого интервала изоляции $[a, b]$ с заданной точностью $\varepsilon = 0.01; 0.001; 0.0001$ найти корни уравнения с использованием метода деления отрезка пополам, метода Ньютона, метода простых итераций.

4. По результатам вычислений составить таблицу для всех значений ε :

Метод решения	Точность	Значение корня	Количество итераций

5. Сделать вывод о точности полученных решений и скорости сходимости изученных методов.

Варианты заданий

Дано уравнение вида $x^3 + a_2x^2 + a_1x + a_0 = 0$

Вариант	a_2	a_1	a_0	Вариант	a_2	a_1	a_0
1	-1	-9	9	15	6	-8.45	-3
2	-8	-16	-6	16	-9.2	15.89	-6.958
3	-7	-13	9	17	4.5	-9	-4.5
4	-6	11	-6	18	-7.6	13	-5
5	2	-12.75	-6.75	19	4.25	-8	-23.75
6	3	1	-1	20	-14.1	1.38	0.28
7	13	10	0	21	-14	34	-7
8	4	-10	4	22	5.4	-4.45	-38.25
9	-1.5	-7	7.5	23	-5.3	-3.45	0
10	-4.5	-8.5	4.5	24	-1	-10	15
11	2	-10	-20	25	-5	4	5.5
12	1	-4.59	0	26	1	-3.5	-4
13	-4.7	7.12	-3.5	27	-10	20	-10
14	1.55	-9.88	2.357	28	-1.7	-3	-0.8

Лабораторная работа № 9

Решение систем линейных алгебраических уравнений

Постановка задачи. Дана система n линейных алгебраических уравнений (СЛАУ) с n неизвестными:

$$\begin{cases} a_{11} \cdot x_1 + a_{12} \cdot x_2 + \dots + a_{1n} \cdot x_n = b_1 \\ a_{21} \cdot x_1 + a_{22} \cdot x_2 + \dots + a_{2n} \cdot x_n = b_2 \\ \dots \\ a_{n1} \cdot x_1 + a_{n2} \cdot x_2 + \dots + a_{nn} \cdot x_n = b_n \end{cases} \quad (4)$$

Перепишем систему в матричном виде:

$$A \cdot \vec{x} = \vec{b}, \quad (5)$$

где $A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$ – матрица размерности $n \times n$,

$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}$ – вектор неизвестных n -го порядка, $\vec{b} = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix}$ – вектор

свободных членов.

Решением СЛАУ (5) называется такая совокупность чисел x_1, x_2, \dots, x_n , которая обращает все уравнения системы (5) в верные равенства. Система (5), определитель которой не равен нулю ($\det A \neq 0$), имеет **единственное решение**.

Использование функций Mathcad для решения СЛАУ

1) **Метод обратной матрицы.** Этот метод относится к классу точных. Точные методы дают решение за конечное число операций. Обратной по отношению к данной называется матрица A^{-1} , которая при умножении на исходную матрицу дает единичную матрицу: $A \cdot A^{-1} = A^{-1} \cdot A = E$. На рис. 13 приведен при-

мер решения СЛАУ методом обратной матрицы, с проверкой решения.

$$\begin{array}{l}
 A := \begin{pmatrix} 6 & -2 & -1 & -1 \\ 1 & 6 & -1 & -2 \\ 5 & -1 & 10 & -3 \\ 0 & 1 & -2 & 5 \end{pmatrix} \quad b := \begin{pmatrix} -5 \\ 2 \\ 21 \\ 16 \end{pmatrix} \quad x := A^{-1} \cdot b = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} \\
 A \cdot x - b = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -3.553 \times 10^{-15} \end{pmatrix} \quad |A \cdot x - b| = 3.768 \times 10^{-15}
 \end{array}$$

Рис. 13. Пример решения СЛАУ методом обратной матрицы

2) **Метод Крамера.** Этот метод относится к классу точных. Решение ищется как отношение определителей вспомогательных матриц к определителю исходной матрицы. Вспомогательные матрицы получаются заменой соответствующего столбца на вектор свободных членов. На рис. 14 приведен пример решения СЛАУ методом Крамера, с проверкой решения.

$$\begin{array}{l}
 \det(A, b, i) := \begin{vmatrix} A^{(i)} \leftarrow b \\ |A| \end{vmatrix} \quad i := 1..rows(A) \quad x_i := \frac{\det(A, b, i)}{|A|} \\
 x = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} \quad A \cdot x - b = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad |A \cdot x - b| = 0
 \end{array}$$

Рис. 14. Пример решения СЛАУ методом Крамера

3) **Функция *Isolve* ().** Для решения СЛАУ можно использовать функцию Mathcad вида *Isolve*(A, b).

4) Блок *Given...Find()*. Перед использованием этого блока требуется задать начальный вектор решения, например нулевой (рис. 15).

$$\begin{array}{l}
 \underline{\underline{A}} := \begin{pmatrix} 6 & -2 & -1 & -1 \\ 1 & 6 & -1 & -2 \\ 5 & -1 & 10 & -3 \\ 0 & 1 & -2 & 5 \end{pmatrix} \quad \mathbf{b} := \begin{pmatrix} -5 \\ 2 \\ 21 \\ 16 \end{pmatrix} \quad \mathbf{x} := \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \\
 \\
 \text{Given} \\
 \\
 \mathbf{A} \cdot \mathbf{x} = \mathbf{b} \\
 \\
 \mathbf{x} := \text{Find}(\mathbf{x}) = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} \quad |\mathbf{A} \cdot \mathbf{x} - \mathbf{b}| = 0
 \end{array}$$

Рис. 15. Использование блока *Given...Find()* в решении СЛАУ

Итерационные методы решения СЛАУ

Итерационные методы являются приближенными, в них решение СЛАУ есть предел последовательных приближений (итераций).

Метод простых итераций. Для расчета по методу простых итераций из каждой i -й строки СЛАУ (4) выражают x_i при условии $a_{ii} \neq 0$, $i = 1, 2, \dots, n$. Задав начальный вектор (например, нулевой), получают формулу

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right), \quad (6)$$

$i = 1, 2, \dots, n$; $k = 0, 1, \dots$ – номер итерации. Итерации прекращаются при выполнении условия $\max_i |x_i^{(k+1)} - x_i^{(k)}| < \varepsilon$, где ε – наперед заданное число, точность расчетов.

Достаточным условием сходимости итерационных вычислений (6) является **условие диагонального преобладания**. Оно означает, что элементы матрицы, расположенные на главной диагонали, по абсолютной величине должны превосходить сумму остальных элементов в соответствующих строках, взятых по абсолютной величине:

$$|a_{ii}| > \sum_{\substack{j=1 \\ i \neq j}}^n |a_{ij}|, \quad i = 1, 2, \dots, n. \quad (7)$$

На рис. 16 показан алгоритм проверки условия (7). На рис. 17 приведен пример решения СЛАУ методом простых итераций.

```

Diagonal(A) := for i ∈ 1..rows(A)
                | s ← 0
                | for j ∈ 1..cols(A)
                |   s ← s + |Ai,j| if i ≠ j
                |   "есть диаг. преобладание" if |Ai,i| > s
                | otherwise
                |   | "нет диаг. преобладания"
                |   | break
Diagonal  $\left( \begin{pmatrix} 4 & 1 & 2 \\ 1 & 0 & -1 \\ -1 & 2 & 7 \end{pmatrix} \right)$  = "нет диаг. преобладания"

```

Рис. 16. Проверка наличия диагонального преобладания

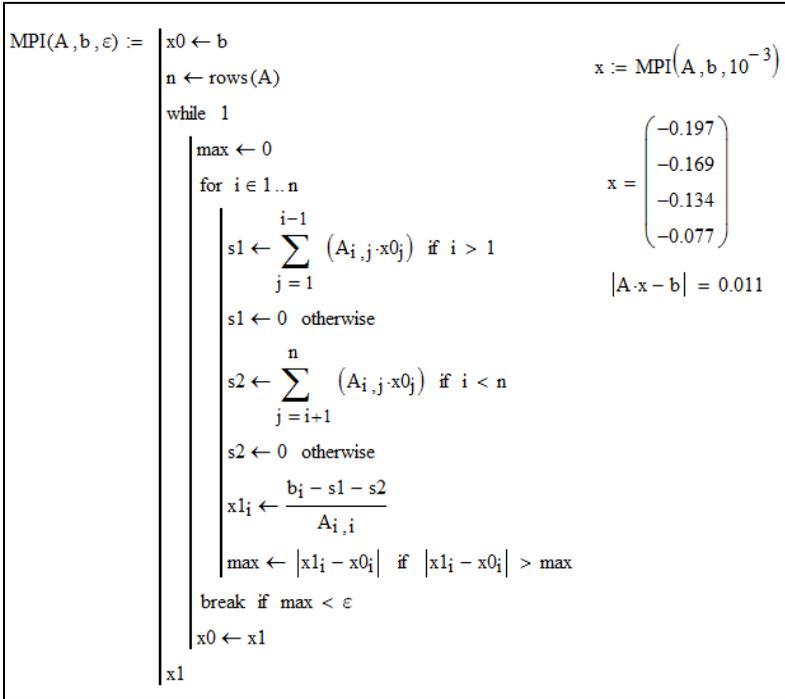


Рис. 17. Пример решения СЛАУ методом простых итераций

Метод Гаусса – Зейделя. Данный метод отличается от метода простых итераций тем, что при расчете элементов вектора решения учитываются вычисления впереди стоящих элементов, выполненные на текущей $k+1$ итерации:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right). \quad (8)$$

Таким образом, алгоритм метода Гаусса – Зейделя отличается от алгоритма метода простых итераций учетом компонентов $x1_j$ в расчете суммы $s1$.

Задание

Решить СЛАУ согласно своему варианту:

а) точными методами обратной матрицы и Крамера;

б) с помощью функции *lsolve()* и блока *Given...Find()*;

в) итерационными методами простых итераций и Гаусса–Зейделя с точностью $\varepsilon = 0.01; 0.001; 0.0001$. Алгоритм изменить так, чтобы в функции считалось число итераций. Вывести погрешность решений. Сравнить итерационные методы по скорости сходимости. Результаты представить в виде таблицы:

Метод	x_1	x_2	x_3	x_4	max	ε	Число итераций
Метод простых итераций						0.01	
						0.001	
						0.0001	
Метод Гаусса–Зейделя						0.01	
						0.001	
						0.0001	

Варианты заданий

1	$A = \begin{pmatrix} 4 & -2 & -1 & 0 \\ -2 & 4 & 0 & 1 \\ -1 & 0 & 4 & 2 \\ 0 & 1 & 2 & 4 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$	2	$A = \begin{pmatrix} 7 & 2 & 2 & 2 \\ 4 & 13 & 4 & 4 \\ 6 & 6 & 15 & 2 \\ 8 & 8 & 8 & 25 \end{pmatrix} \quad b = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.3 \\ 0.4 \end{pmatrix}$
3	$A = \begin{pmatrix} 13 & -4 & -4 & -4 \\ -1 & 4 & -1 & -1 \\ 4 & 4 & 11 & 2 \\ 11 & 11 & 11 & 34 \end{pmatrix} \quad b = \begin{pmatrix} 0.1 \\ 0.15 \\ 0.2 \\ 0.25 \end{pmatrix}$	4	$A = \begin{pmatrix} 24 & -3 & -7 & -13 \\ -1 & 7 & -1 & -4 \\ -1 & 5 & 18 & 11 \\ -1 & 12 & 23 & 37 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 3 \\ 5 \\ 7 \end{pmatrix}$

5	$A = \begin{pmatrix} 27 & -3 & -8 & -15 \\ 3 & 21 & -5 & -12 \\ 8 & 5 & 21 & -7 \\ 15 & 12 & 7 & 35 \end{pmatrix} \quad b = \begin{pmatrix} -5 \\ 0 \\ 5 \\ 10 \end{pmatrix}$	6	$A = \begin{pmatrix} 53 & -6 & -16 & -30 \\ 3 & 44 & -13 & -27 \\ 8 & 22 & 33 & -22 \\ 15 & 9 & -1 & 26 \end{pmatrix} \quad b = \begin{pmatrix} 0.05 \\ 0.2 \\ 0.45 \\ 0.8 \end{pmatrix}$
7	$A = \begin{pmatrix} 36 & -5 & -11 & -19 \\ 1 & 33 & -11 & -20 \\ 5 & -1 & 26 & -19 \\ 11 & 4 & -5 & 21 \end{pmatrix} \quad b = \begin{pmatrix} -9 \\ -8 \\ -7 \\ -6 \end{pmatrix}$	8	$A = \begin{pmatrix} 2.86 & 0.14 & -0.76 & -0.96 \\ 0.14 & 2.38 & -0.96 & -0.28 \\ -0.76 & -0.96 & 3.37 & 0.66 \\ -0.96 & -0.28 & 0.66 & 2.90 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$
9	$A = \begin{pmatrix} 10 & -2 & -3 & -4 \\ 1 & 5 & -1 & -2 \\ 5 & 4 & 12 & 2 \\ 11 & 10 & 9 & 31 \end{pmatrix} \quad b = \begin{pmatrix} 0 \\ -3 \\ -8 \\ -15 \end{pmatrix}$	10	$A = \begin{pmatrix} 10 & -2 & -3 & -4 \\ 1 & 5 & -1 & -2 \\ 5 & 4 & 12 & 2 \\ 11 & 10 & 9 & 31 \end{pmatrix} \quad b = \begin{pmatrix} -1.00 \\ -0.50 \\ -0.33 \\ -0.25 \end{pmatrix}$
11	$A = \begin{pmatrix} 2.49 & 0.98 & 0.43 & 0.08 \\ 0.98 & 2.98 & 0.08 & 0.92 \\ 0.43 & 0.08 & 2.08 & 0.57 \\ 0.08 & 0.92 & 0.57 & 2.57 \end{pmatrix} \quad b = \begin{pmatrix} 0.84 \\ 0.91 \\ 0.14 \\ -0.76 \end{pmatrix}$	12	$A = \begin{pmatrix} 9 & 0 & 2 & 6 \\ 4 & 11 & 2 & 4 \\ 12 & 8 & 27 & 6 \\ 24 & 18 & 14 & 57 \end{pmatrix} \quad b = \begin{pmatrix} 0.67 \\ 0.75 \\ 0.80 \\ 0.83 \end{pmatrix}$
13	$A = \begin{pmatrix} 21 & 0 & 2 & 6 \\ 4 & 19 & 2 & 4 \\ 12 & 8 & 35 & 6 \\ 24 & 18 & 14 & 69 \end{pmatrix} \quad b = \begin{pmatrix} 0.67 \\ 0.75 \\ 0.80 \\ 0.83 \end{pmatrix}$	14	$A = \begin{pmatrix} 5 & -2 & -1 & 0 \\ -2 & 6 & 0 & 1 \\ -1 & 0 & 7 & 2 \\ 0 & 1 & 2 & 8 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$
15	$A = \begin{pmatrix} 9 & 2 & 2 & 2 \\ 4 & 17 & 4 & 4 \\ 6 & 6 & 25 & 6 \\ 8 & 8 & 8 & 33 \end{pmatrix} \quad b = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.3 \\ 0.4 \end{pmatrix}$	16	$A = \begin{pmatrix} 17 & -4 & -4 & -4 \\ -1 & 5 & -1 & -1 \\ 4 & 4 & 17 & 4 \\ 11 & 11 & 11 & 45 \end{pmatrix} \quad b = \begin{pmatrix} 0.10 \\ 0.15 \\ 0.20 \\ 0.25 \end{pmatrix}$

<p>17</p> $A = \begin{pmatrix} 65 & -5 & -11 & -19 \\ 1 & 64 & -11 & -20 \\ 5 & -1 & 57 & -19 \\ 11 & 4 & -5 & 50 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 3 \\ 5 \\ 7 \end{pmatrix}$	<p>18</p> $A = \begin{pmatrix} 51 & -3 & -8 & -15 \\ 3 & 42 & -5 & -12 \\ 8 & 5 & 37 & -7 \\ 15 & 12 & 7 & 44 \end{pmatrix} \quad b = \begin{pmatrix} -5 \\ 0 \\ 5 \\ 10 \end{pmatrix}$
<p>19</p> $A = \begin{pmatrix} 21 & -2 & -4 & -6 \\ 1 & 17 & -3 & -5 \\ 2 & 0 & 13 & -4 \\ 3 & 1 & -1 & 11 \end{pmatrix} \quad b = \begin{pmatrix} 0.05 \\ 0.20 \\ 0.45 \\ 0.80 \end{pmatrix}$	<p>20</p> $A = \begin{pmatrix} 65 & -5 & -11 & -19 \\ 1 & 64 & -11 & -20 \\ 5 & -1 & 57 & -19 \\ 11 & 4 & -5 & 50 \end{pmatrix} \quad b = \begin{pmatrix} -9 \\ -8 \\ -7 \\ -6 \end{pmatrix}$

Лабораторная работа № 10

Часть I. Интерполирование функций

Постановка задачи интерполяции. На интервале $[a, b]$ заданы N значений $x_i, i = 0, 1, \dots, N$; $a \leq x_i \leq b$, называемые **узлами интерполяции**, и значения неизвестной функции в этих узлах $f_i, i = 0, 1, \dots, N$. Могут быть поставлены следующие задачи:

- 1) построить функцию $F(x)$, принимающую в узлах интерполяции x_i заданные значения f_i , т.е. $F(x_i) = f_i, i = 0, 1, \dots, N$;
- 2) для заданного значения $z \in [a, b]$ найти $F(z)$, при этом z не является узлом интерполяции.

Локальные методы интерполяции сводятся к построению функции $F_i(x)$ на каждом из частичных отрезков $[x_{i-1}, x_i], i = 1, 2, \dots, N$.

В **кусочно-постоянных методах** функция заменяется константой:

для **кусочно-постоянной интерполяции слева**

$$F_i(x) = f_{i-1}, \quad x_{i-1} \leq x \leq x_i, \quad (9)$$

для **кусочно-постоянной интерполяции справа**

$$F_i(x) = f_i, \quad x_{i-1} \leq x \leq x_i. \quad (10)$$

В **кусочно-линейной интерполяции** на каждом из отрезков $[x_{i-1}, x_i]$ функция заменяется прямой:

$$F_i(x) = k_i x + l_i, \quad (11)$$

где $k_i = \frac{f_i - f_{i-1}}{x_i - x_{i-1}}, \quad l_i = f_i - k_i x_i, \quad i = 1, 2, \dots, N$.

На рис. 18 приведен пример решения задачи методом кусочно-постоянной интерполяции слева, на рис. 19 – методом кусочно-линейной интерполяции.

Для проведения **гладких** кривых через узлы интерполяции используют **кусочно-полиномиальную интерполяцию сплайнами**.

```

ORIGIN := 0
ConstLeft(x, y, z) := | n ← length(x) - 1
                      | return "error" if z < x0
                      | return yn if z ≥ xn
                      | for i ∈ 1..n
                      |   return yi-1 if xi-1 ≤ z < xi

```

$$x := \begin{pmatrix} 0 \\ 1.5 \\ 3.3 \\ 4.85 \\ 5.3 \end{pmatrix} \quad y := \begin{pmatrix} -2 \\ 3.5 \\ 1.7 \\ 4 \\ 6 \end{pmatrix}$$

```

ConstLeft(x, y, -1) = "error"   ConstLeft(x, y, 5.8) = 6
z := 4       ConstLeft(x, y, z) = 1.7

```

Рис. 18. Пример решения задачи методом кусочно-постоянной интерполяции слева

```

ORIGIN := 0
Linear(x, y, z) := | n ← length(x) - 1
                  | return y0 if z < x0
                  | return yn if z > xn
                  | il ← 0
                  | for i ∈ 1..n
                  |   if xi-1 ≤ z ≤ xi
                  |     k ←  $\frac{y_{i-1} - y_i}{x_{i-1} - x_i}$ 
                  |     l ← yi - k · xi
                  |     return k · z + l
                  |   break

```

$$x := \begin{pmatrix} 0 \\ 1.5 \\ 3.3 \\ 4.85 \\ 5.3 \end{pmatrix} \quad y := \begin{pmatrix} -2 \\ 3.5 \\ 1.7 \\ 4 \\ 6 \end{pmatrix}$$

```

Linear(x, y, -1) = -2           Linear(x, y, 5.8) = 6           Linear(x, y, 3.3) = 1.7
z := 4       Linear(x, y, z) = 2.739

```

Рис. 19. Пример решения задачи методом кусочно-линейной интерполяции

Сплайн – это функция, которая на каждом частичном отрезке представляет собой полином, а на всем отрезке $[a, b]$ непрерывна вместе с производными нескольких порядков. Например, в **кубических интерполяционных сплайнах** используются полиномы третьей степени, имеющие непрерывные производные первого и второго порядка. При расчетах сплайнами необходимо задать **граничные условия** (описать поведение сплайна) в точках x_0 и x_N .

Функции кубических сплайнов в Mathcad

1) **Функция *interp*()**. Функция *interp*(s, x, y, z) вычисляет значения функции в точке z , где x и y – векторы исходных данных, причем вектор x упорядочен по возрастанию; z – значение аргумента, при котором вычисляется сплайн; s – вектор вторых производных от табличных значений (x, y).

2) **Функция *lspline*()**. Функция *lspline*(x, y) формирует вектор s исходя из **линейных** краевых условий.

3) **Функция *pspline*()**. Функция *pspline*(x, y) формирует вектор s исходя из **параболических** краевых условий.

4) **Функция *cspline*()**. Функция *cspline*(x, y) формирует вектор s исходя из **кубических** краевых условий.

На рис. 20 показан пример использования функций Mathcad при интерполяции сплайнами.

Глобальные методы означают, что на отрезке $[a, b]$ подбирается **единая** функция $F(x)$ такая, что $F(x_i) = f_i, i = 0, 1, \dots, N$. Пусть интерполирующая функция $F(x)$ – это полином степени N . Будем использовать интерполяционный полином Лагранжа:

$$\text{Lagrange}(x, y, z) := \sum_{i=0}^{\text{length}(y)-1} \left(y_i \cdot \prod_{j=0}^{\text{length}(x)-1} \text{if} \left(j \neq i, \frac{z - x_j}{x_i - x_j}, 1 \right) \right) \quad (12)$$

```

x :=  $\begin{pmatrix} 0 \\ 1.5 \\ 3.3 \\ 4.85 \\ 5.3 \end{pmatrix}$    y :=  $\begin{pmatrix} -2 \\ 3.5 \\ 1.7 \\ 4 \\ 6 \end{pmatrix}$    z := 4

sl := lspline(x, y)
x1 := interp(sl, x, y, z)   x1 = 1.827
sp := pspline(x, y)
xp := interp(sp, x, y, z) = 1.896
sc := cspline(x, y)   xc := interp(sc, x, y, z) = 1.926

```

Рис. 20. Пример использования функций Mathcad при интерполяции сплайнами

Функции Mathcad для решения задачи интерполяции глобальными методами

1) **Функция *polyint* ()**. Функция *polyint*(*x*, *y*, *z*) находит полином степени $N-1$, проходящий через каждую точку $\{x_i, y_i\}$. Результатом вычислений функции является вектор из двух элементов – найденное в точке *z* значение полинома $F(z)$ и оценка погрешности решения.

2) **Функция *polycoeff* ()**. Функция *polycoeff*(*x*, *y*) для интерполяционного полинома $F(x) = c_0 + c_1x^1 + \dots + c_Nx^N$ вычисляет коэффициенты c_i . Результатом вычислений функции является вектор этих коэффициентов размерности $N + 1$. Окончательно решение задачи интерполяции в точке *z* находится так:

$$F(z) = \sum_{i=0}^N c_i \cdot z^i.$$

3) **Функция *polyiter*()**. Функция *polyiter*(x, y, z, N, ε) позволяет задать допустимую погрешность решения ε , что обычно приводит к снижению степени полинома. Результатом вычислений функции является вектор, в котором вторая и третья компоненты – это степень полинома и его значение в точке z .

$$\begin{array}{l}
 x := \begin{pmatrix} 0 \\ 1.5 \\ 3.3 \\ 4.85 \\ 5.3 \end{pmatrix} \quad y := \begin{pmatrix} -2 \\ 3.5 \\ 1.7 \\ 4 \\ 6 \end{pmatrix} \quad N := \text{length}(x) - 1 \quad z := 4 \\
 \\
 \text{polyint}(x, y, z) = \begin{pmatrix} 1.917 \\ 0.281 \end{pmatrix} \\
 \\
 c := \text{polycoeff}(x, y) \quad c = \begin{pmatrix} -2 \\ 9.12 \\ -4.879 \\ 0.9 \\ -0.047 \end{pmatrix} \quad \sum_{i=0}^N (c_i \cdot z^i) = 1.917 \\
 \\
 \varepsilon := 10^{-2} \quad \text{polyiter}(x, y, z, N, \varepsilon) = \begin{pmatrix} 0 \\ 4 \\ 1.917 \end{pmatrix}
 \end{array}$$

Рис. 21. Пример использования функций Mathcad в решении интерполяционных задач

Задание

Даны исходная функция $g(x)$ и отрезок $[a, b]$.

1. Вычислить узлы интерполяции $x_i = a + h \cdot i$, $h = \frac{b-a}{N}$, $i = 0, 1, \dots, N$ (N – параметр задачи, $N = 10$), и найти в этих узлах значения $y_i = g(x_i)$. Вывести на экран векторы x и y .

2. Построить график функции $g(x)$ на отрезке $[a, b]$.

3. Задать значение z в некоторой произвольной точке на отрезке $[a, b]$, не совпадающей ни с одним из узлов x_i . Вычислить значение $F(z)$ с помощью кусочно-постоянной слева, кусочно-постоянной справа и кусочно-линейной интерполяции. Для кусочно-линейной интерполяции выписать уравнение прямой $F(x) = kx + l$. Найти погрешность $|g(z) - F(z)|$ для каждого метода интерполяции.

4. Вычислить значение $F(z)$ с помощью полинома Лагранжа (12). Найти погрешность метода $|g(z) - F(z)|$. Расчеты провести при $N = 10$ и $N = 20$.

5. Вычисления оформить в виде таблицы:

Метод интерполяции	z	$g(z)$	$F(z)$	$ g(z) - F(z) $
Кусочно-постоянная слева				
Кусочно-постоянная справа				
Кусочно-линейная				
Полином Лагранжа, $N = 10$				
Полином Лагранжа, $N = 10$				

Варианты заданий

Вариант	$g(x)$	$[a, b]$	Вариант	$g(x)$	$[a, b]$
1	$\frac{2}{1-x^2}$	$[2, 3]$	15	$2^{-x} - 1$	$[1, 2]$
2	$\frac{1-x}{1+x^2}$	$[-1, 1]$	16	$\frac{1}{\sin(x)^2}$	$\left[\frac{\pi}{4}, \frac{3\pi}{4}\right]$
3	$\frac{1}{1+x^2}$	$[2, 3]$	17	$\frac{1}{\cos(x)^2}$	$\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$
4	$\frac{2-x^2}{3-x}$	$[-1, 0.5]$	18	$\frac{1-x}{x}$	$[-1, 0.5]$
5	$\sin(2x)$	$[0, 1]$	19	$\sin(x) + \cos(2x)$	$\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$
6	$x + \sqrt{x} + \sqrt[3]{x}$	$[0, 2]$	20	$x^{1/3} + x^{-2/3}$	$[1, 2]$
7	$\frac{1}{\sin(x)^2}$	$[0.5, 1.5]$	21	$\frac{5}{\sin(x)^2}$	$\left[\frac{\pi}{4}, \frac{\pi}{2}\right]$
8	$\frac{3}{\sqrt{1-x^2}}$	$\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$	22	$e^{x/3}$	$[-0.5, 0.5]$
9	e^x	$[-1, 1]$	23	$\sin(x) - 3\cos(2x)$	$\left[\frac{\pi}{4}, \frac{\pi}{2}\right]$
10	$x^{2/3}$	$[0, 5]$	24	$\frac{1}{1+x} + \frac{2}{1-x^2}$	$[-0.5, 0.5]$
11	$\cos(x) - \frac{1}{2}\cos(2x)$	$[-\pi, \pi]$	25	$\cos(x) - \frac{1}{4}\sin(2x)$	$[-\pi, \pi]$
12	$\sin(x) + \frac{1}{3}\sin(3x)$	$[0, 2\pi]$	26	$\frac{3}{\cos(x)^2}$	$\left[0, \frac{\pi}{2}\right]$
13	$\frac{1}{1+x}$	$[1, 3]$	27	$\frac{5}{\sqrt{1-x^2}}$	$[2, 3]$
14	$x^3 - 3x^2 + 2$	$[0, 1]$	28	$\frac{x^2 - 3x + 2}{1+x^2}$	$[0, 1]$

Лабораторная работа № 10

Часть II. Аппроксимация функций

Постановка задачи аппроксимации. Пусть опытным путем в результате измерений на интервале $[a, b]$ получено $N + 1$ значение x_i и y_i , $i = 0, 1, \dots, N$. Требуется построить полиномы:

- а) первой степени $P_1(x) = a_1 + a_2x$;
- б) второй степени $P_2(x) = a_1 + a_2x + a_3x^2$;
- в) третьей степени $P_3(x) = a_1 + a_2x + a_3x^2 + a_4x^3$.

Неизвестные коэффициенты a_i находятся из следующего условия: полиномы должны быть самыми близкими к заданным точкам (x_i, y_i) из всех возможных полиномов. Степень близости к эмпирическим данным определяется в смысле **метода наименьших квадратов (МНК)**, т.е. из условия минимума суммы квадратов отклонений вычисленных (аппроксимирующих) и эмпирических данных:

$$\sum_{i=0}^N (y_i - P_k(x_i))^2 \rightarrow \min. \quad (13)$$

Вектор неизвестных коэффициентов a полинома $P_3(x)$ находится из решения системы нормальных уравнений вида $B \cdot a = f$,

$$B = \begin{pmatrix} N+1 & \sum_{i=0}^N x_i & \sum_{i=0}^N x_i^2 & \sum_{i=0}^N x_i^3 \\ \sum_{i=0}^N x_i & \sum_{i=0}^N x_i^2 & \sum_{i=0}^N x_i^3 & \sum_{i=0}^N x_i^4 \\ \sum_{i=0}^N x_i^2 & \sum_{i=0}^N x_i^3 & \sum_{i=0}^N x_i^4 & \sum_{i=0}^N x_i^5 \\ \sum_{i=0}^N x_i^3 & \sum_{i=0}^N x_i^4 & \sum_{i=0}^N x_i^5 & \sum_{i=0}^N x_i^6 \end{pmatrix}, \quad f = \begin{pmatrix} \sum_{i=0}^N y_i \\ \sum_{i=0}^N y_i \cdot x_i \\ \sum_{i=0}^N y_i \cdot x_i^2 \\ \sum_{i=0}^N y_i \cdot x_i^3 \end{pmatrix}. \quad (14)$$

Тогда методом обратной матрицы получим $a = B^{-1} \cdot f$. Для полиномов $P_1(x)$ и $P_2(x)$ матрица B имеет размерность 2×2 и 3×3 соответственно, вектор f состоит из двух и трех элементов соответственно.

Функции Mathcad для аппроксимации функций

1) Функция *line*(). Функция *line*(x, y) для исходных векторов x, y вычисляет коэффициенты a_0 и a_1 уравнения прямой $P_1(x)$.

2) Функция *regress*(). Функция *regress*(x, y, m) для исходных векторов x, y и заданной степени полинома m вычисляет коэффициенты этого полинома (для прямой – это последние два элемента вектора решения, для полинома второй степени – последние три элемента, для полинома третьей степени – последние четыре элемента).

Пример

Для заданных векторов x и y найти аппроксимирующую функцию – полином первой степени $P_1(x)$, вычислить погрешность аппроксимации (рис. 22). Проверить решение на функциях Mathcad.

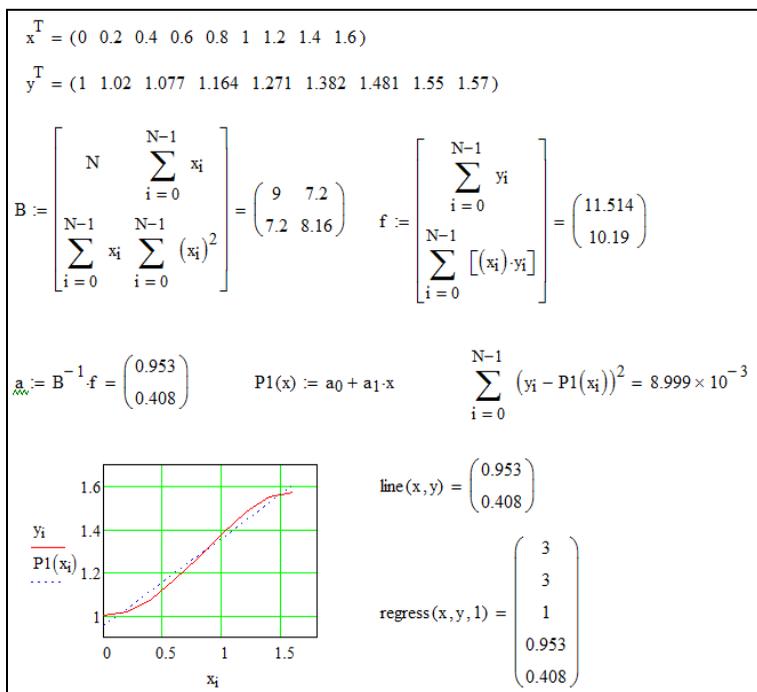


Рис. 22. Подбор аппроксимирующей функции

Задание

1. Получить систему нормальных уравнений для полиномов $P_1(x)$, $P_2(x)$, $P_3(x)$. Вычислить коэффициенты a_i для каждого полинома.

2. Определить, какой из полиномов имеет минимальную сумму квадратов отклонений.

3. На одном рисунке построить графики исходных данных и найденных уравнений.

4. Проверить решение с помощью встроенных функций *line* и *regress*.

Варианты заданий

Вариант	x													
	-10	-8.3	-6.7	-5	-3	-1.67	0	1.67	3.33	5	6.7	8.33	10	
1	0.02	0.19	0.4	1	1	0.85	1.02	1.19	1.35	1.5	1.7	1.85	2	
2	21	14.9	9.9	6	3	1.56	1	1.56	3.22	6	9.9	14.9	21	
3	-310	-182	-96	-43	-14	-3.06	0	3.06	14.4	43	96	182	310	
4	-2.98	-2.3	-1.6	-1	0	0.35	1.02	1.69	2.35	3	3.7	4.35	5	
5	51	35.7	23	14	7	2.39	1	2.39	6.56	14	23	35.7	51	
6	-610	-356	-184	-80	-26	-4.44	0	4.44	25.6	80	184	356	610	
7	-5.98	-4.8	-3.6	-2	-1	-0.15	1.02	2.19	3.35	4.5	5.7	6.85	8	
8	81	56.6	37	21	10	3.22	1	3.22	9.89	21	37	56.6	81	
9	-100	-60	-33	-16	-7	-2.08	0	2.08	6.67	16	33	60.4	100	
10	-8.98	-7.3	-5.6	-4	-2	-0.65	1.02	2.69	4.35	6	7.7	9.35	11	
11	56	39.2	25	15	7	2.53	1	2.53	7.11	15	25	39.2	56	
12	-70	-43	-24	-13	-6	-1.94	0	1.94	5.56	13	24	43.1	70	
13	-11.98	-9.8	-7.6	-5	-3	-1.15	1.02	3.19	5.35	7.5	9.7	11.9	14	
14	71	49.6	32	19	9	2.94	1	2.94	8.78	19	32	49.6	71	
15	-85	-52	-29	-14	-6	-2.01	0	2.01	6.11	14	29	51.7	85	
16	-14.98	-12	-9.6	-7	-4	-1.65	1.02	3.69	6.35	9	12	14.4	17	
17	86	60	39	22	10	3.36	1	3.36	10.4	22	39	60	86	
18	-70	-43	-24	-13	-6	-1.94	0	1.94	5.56	13	24	43.1	70	
19	-17.98	-15	-12	-8	-5	-2.15	1.02	4.19	7.35	11	14	16.9	20	
20	51	35.7	23	14	7	2.39	1	2.39	6.56	14	23	35.7	51	
21	-80	-49	-27	-14	-6	-1.99	0	1.99	5.93	14	27	48.8	80	
22	-20.98	-17	-14	-10	-6	-2.65	1.02	4.69	8.35	12	16	19.4	23	
23	58.5	40.9	27	15	7	2.6	1	2.6	7.39	15	27	40.9	59	
24	-90	-55	-30	-15	-6	-2.04	0	2.04	6.3	15	30	54.6	90	
25	-23.98	-20	-16	-11	-7	-3.15	1.02	5.19	9.35	14	18	21.9	26	

Лабораторная работа № 11 Численное интегрирование

Постановка задачи. Вычислить значение определенного интеграла

$$I_1 = \int_a^b f(x)dx, \quad (15)$$

где $f(x)$ – непрерывная на отрезке $[a, b]$ функция.

Для приближенного вычисления интеграла используются **квадратурные формулы**. Для этого отрезок $[a, b]$ разбивается на N равных отрезков длиной h . Величина $h = x_i - x_{i-1}$, $i = 1, 2, \dots, N$, называется **шагом интегрирования** и вычисляется как $h = \frac{b-a}{N}$. Далее на каждом из частичных отрезков $[x_{i-1}, x_i]$ подбирается функция $f(x)$. Тогда приближенное значение интеграла (15) вычисляется как

$$I_N = \sum_{i=0}^N c_i \cdot f(x_i). \quad (16)$$

Метод левых прямоугольников. Здесь функция $f(x) = f(x_i)$, $x \in [x_i, x_{i+1}]$, и

$$\int_a^b f(x)dx \approx h \sum_{i=0}^{N-1} f(x_i). \quad (17)$$

Метод правых прямоугольников. Здесь функция $f(x) = f(x_i)$, $x \in [x_{i-1}, x_i]$, и

$$\int_a^b f(x)dx \approx h \sum_{i=1}^N f(x_i). \quad (18)$$

Оба этих метода имеют первый по h порядок точности.

Метод средних прямоугольников. Квадратурная формула имеет вид:

$$\int_a^b f(x)dx \approx h \sum_{i=1}^N f\left(\frac{x_{i-1} + x_i}{2}\right). \quad (19)$$

Метод трапеций. Квадратурная формула имеет вид:

$$\int_a^b f(x)dx \approx h \left(\frac{f(a) + f(b)}{2} + \sum_{i=1}^N f(x_i) \right). \quad (20)$$

Метод средних прямоугольников и метод трапеций имеют второй по h порядок точности.

Формула Симпсона. Обозначим $f_i = f(x_i)$, $f_{i-1/2} = f\left(x_{i-1/2}\right)$, $f_{i-1} = f(x_{i-1})$. Тогда

$$\int_a^b f(x)dx \approx \frac{h}{6} \sum_{i=1}^N \left[f(x_{i-1}) + 4f\left(\frac{x_{i-1} + x_i}{2}\right) + f(x_i) \right]. \quad (21)$$

Метод имеет четвертый по h порядок точности.

Задание

1. Вычислить точное значение интеграла $I_1 = \int_a^b f(x)dx$ для своего варианта.

2. Найти $I_N = \int_a^b f(x)dx$ при $N = 10$, $N = 20$, $N = 40$ по формулам:

мулам:

- правых прямоугольников,
 - средних прямоугольников,
 - трапеций,
 - Симпсона.
3. Вычислить погрешности. Результаты вычислений оформить в виде таблицы.
4. Сделать вывод о порядке аппроксимации квадратурных формул.

Пример

Для функции $f(x) = \sqrt{x^4 - x^3 + 8}$ вычислить значение интеграла на отрезке $[0, 3.2]$ методом левых прямоугольников для $N = 10$.

Найти погрешность решения относительно точного значения.

На рис. 23 показано решение данной задачи. Приведены два варианта ее решения – с помощью модульного и безмодульного программирования. Для каждого случая вычислена погрешность решения. Значительная величина ошибки решения объясняется низкой точностью используемого метода.

$f(x) := \sqrt{x^4 - x^3 + 8}$ $a := 0$ $b := 3.2$ $N := 10$

$I := \int_a^b f(x) dx$ $I = 13.432$

Метод левых прямоугольников.
 Модульное программирование

$pr(a, b, N, f) := \begin{cases} I \leftarrow 0 \\ h \leftarrow \frac{b - a}{N} \\ \text{for } i \in 0..N - 1 \\ \quad I \leftarrow I + f(a + i \cdot h) \\ \quad I \cdot h \end{cases}$ $I_1 := pr(a, b, N, f) = 12.5$

$|I - I_1| = 0.932$

Безмодульное программирование:

$h := \frac{b - a}{N}$ $I_2 := h \cdot \sum_{i=0}^{N-1} f(a + i \cdot h)$ $|I - I_2| = 0.932$

Рис. 23. Приближенное вычисление интеграла методом левых прямоугольников

Варианты заданий

Вариант	$g(x)$	$[a, b]$	Вариант	$g(x)$	$[a, b]$
1	$\frac{2}{1-x^2}$	[2, 3]	15	$2^{-x} - 1$	[1, 2]
2	$\frac{1-x}{1+x^2}$	[-1, 1]	16	$\frac{1}{\sin^2(x)}$	$\left[\frac{\pi}{4}, \frac{3\pi}{4}\right]$
3	$\frac{1}{1+x^2}$	[2, 3]	17	$\frac{1}{\cos^2(x)}$	$\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$
4	$(2-x^2)(3-x)$	[-1, 0.5]	18	$\frac{1-x}{x}$	[-1, 0.5]
5	$\sin(2x)$	[0, 1]	19	$\sin(x) + \cos(2x)$	$\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$
6	$x + \sqrt{x} + \sqrt[3]{x}$	[0, 2]	20	$x^{1/3} + x^{-2/3}$	[1, 2]
7	$\frac{1}{\sin^2(x)}$	$\left[\frac{1}{2}, \frac{3}{2}\right]$	21	$\frac{5}{\sin^2(x)}$	$\left[\frac{\pi}{4}, \frac{\pi}{2}\right]$
8	$\frac{3}{\sqrt{1-x^2}}$	$\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$	22	e^{x^3}	$\left[-\frac{1}{2}, \frac{1}{2}\right]$
9	e^x	[-1, 1]	23	$\sin(x) - 3\cos(2x)$	$\left[\frac{\pi}{4}, \frac{\pi}{2}\right]$
10	$x^{2/3}$	[0, 5]	24	$\frac{1}{1+x} + \frac{2}{1-x^2}$	$\left[-\frac{1}{2}, \frac{1}{2}\right]$
11	$\cos(x) - \frac{1}{2}\cos(2x)$	$[-\pi, \pi]$	25	$\cos(x) - \frac{1}{4}\sin(2x)$	$[-\pi, \pi]$
12	$\sin(x) + \frac{1}{3}\sin(3x)$	[0, 2 π]	26	$\frac{3}{\cos^2(x)}$	$\left[\frac{\pi}{6}, \frac{\pi}{3}\right]$
13	$\frac{1}{1+x}$	[1, 3]	27	$\frac{5}{\sqrt{1+x^2}}$	[2, 3]
14	$x^3 - 3x^2 + 2$	[0, 1]	28	$\frac{x^2 - 3x + 2}{1+x^2}$	[0, 1]

Лабораторная работа № 12
Численное решение обыкновенных дифференциальных уравнений (задача Коши)

Постановка задачи. Дано обыкновенное дифференциальное уравнение (ОДУ) $\frac{dy}{dx} = f(x, y)$. Требуется найти решение $y(x)$ на отрезке $[x_0, x_n]$ при условии $y(x_0) = y_0$.

Разобьем отрезок $[x_0, x_n]$ на n равных отрезков длиной h . Величина $h = x_i - x_{i-1}$, $i = 1, 2, \dots, n$, называется **шагом сетки** и вычисляется как $h = \frac{x_n - x_0}{n}$. Значения $x_i = x_0 + ih$, $i = 0, 1, \dots, n$, называются **узлами сетки**. Необходимо найти приближенные значения в узлах сетки $y_i = y(x_i)$.

Метод Эйлера. Для решения используется рекуррентная формула вида

$$y_{i+1} = y_i + hf(x_i, y_i) \quad (22)$$

с начальным условием $y(x_0) = y_0$, $i = 0, 1, \dots, n - 1$. Метод имеет первый по h порядок точности.

Модифицированный метод Эйлера (метод «предиктор-корректор»). Метод имеет второй по h порядок точности. Для решения используется двухшаговая вычислительная схема вида

$$\begin{aligned} \tilde{y}_{i+1} &= y_i + hf(x_i, y_i) \\ y_{i+1} &= y_i + \frac{h}{2} [f(x_i, y_i) + f(x_{i+1}, \tilde{y}_{i+1})], \quad y(x_0) = y_0, \quad i = 0, 1, \dots, n - 1 \end{aligned} \quad (23)$$

Методы Рунге–Кутты. Рассмотренные выше метод Эйлера и модифицированный метод Эйлера относятся к семейству методов Рунге–Кутты. В библиотеке функций Mathcad есть функция для решения ОДУ четырехшаговым методом Рунге–Кутты вида $rkfixed(y, a, b, n, f)$, где $a = x_0$, $b = x_n$. Данная функция позволяет получить решение с четвертым по h порядком точности.

Пример

Используя метод Эйлера, модифицированный метод Эйлера и метод Рунге – Кутты, решить уравнение $\frac{dy}{dx} = f(x, y)$ на отрезке $[0, 1]$, где $f(x, y) = y(1 - x)$, начальное условие $y(0) = 1$, $N = 8$.

Известно точное решение $u(x) = e^{x - \frac{1}{2}x^2}$. Построить графики точного и найденного решений, вычислить погрешность решения. На рис. 24–26 показан пример решения данной задачи различными методами.

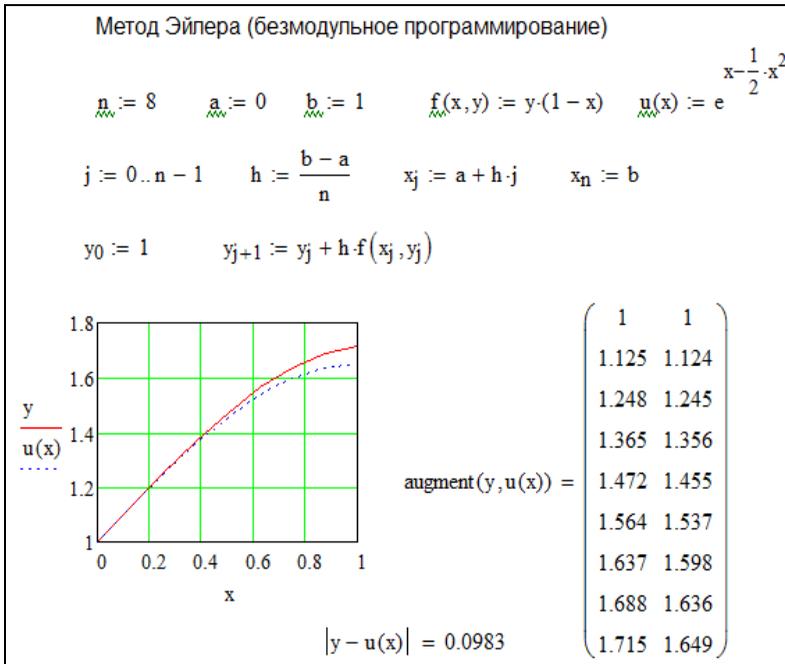


Рис. 24. Решение ОДУ методом Эйлера (безмодульное программирование)

Модифицированный метод Эйлера ("предиктор-корректор")

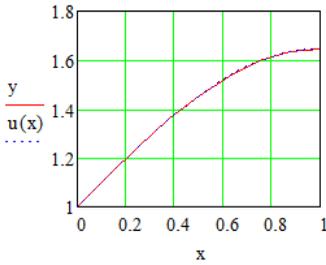
```

M_Eyler(f, y0, a, b, n) :=
  x0 ← a
  y0 ← y0
  h ← (b - a) / n
  for j ∈ 0..n - 1
    xj+1 ← xj + h
    y1j+1 ← yj + h * f(xj, yj)
    yj+1 ← yj + (h / 2) * (f(xj, yj) + f(xj+1, y1j+1))
  augment(x, y)
  
```

$\underline{n} := 8$ $\underline{a} := 0$ $\underline{b} := 1$ $y_0 := 1$ $\underline{f}(x, y) := y \cdot (1 - x)$

$Z := M_Eyler(f, y_0, a, b, n)$

$j := 0..n$ $x_j := Z_{j,0}$ $y_j := Z_{j,1}$ $\underline{u}(x) := e^{\frac{1}{2}x^2}$



$\text{augment}(y, u(x)) =$

1	1
1.124	1.124
1.244	1.245
1.355	1.356
1.454	1.455
1.536	1.537
1.597	1.598
1.635	1.636
1.647	1.649

$|y - u(x)| = 0.0029$

Рис. 25. Решение ОДУ модифицированным методом Эйлера

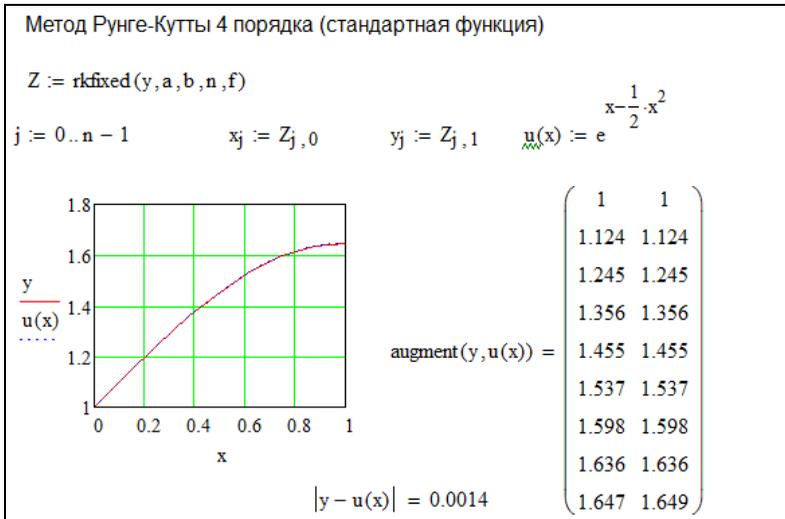


Рис. 26. Решение ОДУ методом Рунге – Кутты (использование стандартной функции)

Задание

1. Для своего варианта проверить, что заданная функция $u(x)$ является точным решением, удовлетворяющим начальному условию $u(a) = \varphi$.

2. Найти решение задачи Коши для ОДУ первого порядка $\frac{dy}{dx} = f(x, y)$ на отрезке $[a, b]$ при $y(x_0) = \varphi$, $N = 10$, $N = 20$ и $N = 40$:

- методом Эйлера,
- модифицированным методом Эйлера,
- методом Рунге – Кутты четвертого порядка.

3. На основании анализа погрешностей определить порядки аппроксимации каждого метода.

Варианты заданий

Вариант	$f(x, y)$	$[a, b]$	φ	$u(x)$
1	$2x - y + x^2$	$[0, 2]$	0	x^2
2	$y/x - 4/x^2$	$[1, 3]$	2	$2/x$
3	$\sin^2(x) - y^2 - \cos(x)$	$\left[0, \frac{\pi}{2}\right]$	0	$-\sin(x)$
4	$0.5(\cos^2(x) - y^2) - \sin(x)$	$\left[0, \frac{\pi}{2}\right]$	1	$\cos(x)$
5	$3x - 0.5xy + x^3$	$[0, 2]$	-2	$2(x^2 - 1)$
6	$x(2 + x^2 \sin(x^2) - y \sin(y))$	$[-2, 0]$	4	x^2
7	$(x+1)(\cos(x) - y) - \sin(x)$	$\left[\frac{\pi}{2}, \frac{3\pi}{2}\right]$	0	$\cos(x)$
8	$2y \cdot \operatorname{ctg}(x) - (y - \sin^2(x))$	$\left[\frac{\pi}{2}, \frac{3\pi}{2}\right]$	1	$\sin^2(x)$
9	$y/x - 8/x^2$	$[2, 5]$	2	$\frac{4}{x}$
10	$\cos^2(x) - y^2 - \sin(x)$	$\left[0, \frac{\pi}{2}\right]$	1	$\cos(x)$
11	$\cos^2(x) - y^2/4 - 2\sin(x)$	$\left[0, \frac{\pi}{2}\right]$	2	$2\cos(x)$
12	$x^3 - y + 3x^2$	$[0, 2]$	0	x^3
13	$(x+1)\sin(x) - y - x\cos(x)$	$[0, \pi]$	0	$x \sin(x)$
14	$x - y^2 + x^4/4$	$[0, 2]$	0	$\frac{x^2}{2}$
15	$x(2 + x^2 \sin(x^2) - y \sin(y))$	$[-2, 0]$	4	x^2
16	$y/x - 8/x^2$	$[1, 3]$	4	$\frac{4}{x}$
17	$0.5(\sin^2(x) - y^2) + \cos(x)$	$\left[0, \frac{\pi}{2}\right]$	0	$\sin(x)$
18	$\cos^2(x) - y^2 - \sin(x)$	$\left[0, \frac{\pi}{2}\right]$	1	$\cos(x)$
19	$3x^2 - y^2 + x^6$	$[0, 2]$	0	x^3
20	$(x+0.6)(\cos(x) - y) - \sin(x)$	$\left[\frac{\pi}{2}, \frac{3\pi}{2}\right]$	0	$\cos(x)$
21	$(x+1)(\sin(x) - y) + \cos(x)$	$\left[\frac{\pi}{2}, \frac{3\pi}{2}\right]$	1	$\sin(x)$
22	$2(x-2) + x(x-2)^2 - xy$	$[2, 4]$	0	$(x-2)^2$

Контрольная работа

Контрольная работа выполняется студентом согласно номеру варианта. Номер варианта – последняя цифра номера студента в журнале преподавателя (0 – вариант № 10).

1. Для функций $f(x)$ и $g(x)$ найти неопределенный интеграл и уравнение производной.
2. Построить графики функций $f(x)$ и $g(x)$ (на одном рисунке).
3. Для функции $f(x)$ вычислить определенный интеграл на отрезке $[a, b]$.
4. Для функции $g(x)$ вычислить первую и вторую производную в точках $x = 2$ и $x = 4$.
5. Для функции $g(x)$ выполнить разложение в ряд Тейлора. Полученное уравнение обозначить как $z(x)$.
6. Для функции $z(x)$ вычислить пределы при $x \rightarrow +\infty$ и $x \rightarrow -\infty$.
7. С помощью функции `polyroots()` найти корни уравнения $f(x)$. Проверить правильность вычисления.
8. Найти корни уравнения $g(x)$ с помощью функции `root()` и блока `Given...Find()`. Если решение отсутствует, найти минимальную ошибку решения с помощью блока `Given...Minerr()`.
9. Решить систему из двух уравнений $f(x)$ и $g(x)$ с помощью блока `Given...Find()`. В качестве стартовой точки выбрать $x = 1, y = 1$.
10. Исследовать функцию $f(x)$ на минимум, функцию $g(x)$ – на максимум. Для этого:
 - приравнять нулю первую производную, найти решение данного уравнения. Полученное решение является точкой экстремума исследуемой функции;
 - для определения – является точка экстремума максимумом или минимумом – вычислить значение функции слева и справа от экстремума и сравнить вычисленные значения со значением функции в точке экстремума;
 - другой вариант – вычислить значение *производной* в точках слева и справа от экстремума. По знаку производных определить участки возрастания и убывания функции;

- сделать вывод о наличии максимума или минимума.
11. Составить векторы из значений $x = 0, 0.1, \dots, 1$ и $y = g(x)$. Записать полученные векторы в текстовый файл. Открыть файл и проверить правильность данных.

Варианты заданий

Вариант	$f(x)$	$g(x)$
1	$(1+x)^2$	$ax^3 + bx^2$
2	$(1-x)^2$	$\sin(ax)$
3	$(a+x)^2$	$\cos(ax)$
4	$(a-x)^2$	$\sec(x)$
5	$(a+x)^3$	$x \ln(1+x)$
6	$(a-bx)^3$	$\frac{1}{(1+x)^2}$
7	$(1+x)^4$	$\frac{1}{(a+bx)}$
8	$(1-x)^4$	$\frac{1}{(1-x)^2}$
9	$(a+bx)^4$	$-\frac{\cos^3(x)}{3}$
10	$(a-x)^4$	$\frac{\sin^3(x)}{3}$

Для всех вариантов
 $a = 1, b = 2.$

Контрольная работа распечатывается на листах формата А4. Файлы Mathcad сохраняются до окончания проверки. На первой странице указывается фамилия студента, номер группы, номер варианта. Пункты заданий нумеруются.

Список литературы

1. Основы вычислений и программирования в пакете MathCAD : учеб. пособие / Ю. Е. Воскобойников [и др.] ; под ред. Ю. Е. Воскобойникова ; Новосиб. гос. архитектур.-строит. ун-т (Сибстрин). – Новосибирск : НГАСУ (Сибстрин), 2012. – 213 с.
2. Решение инженерных задач в пакете MathCAD : учеб. пособие / Ю. Е. Воскобойников [и др.] ; под ред. Ю. Е. Воскобойникова ; Новосиб. гос. архитектур.-строит. ун-т (Сибстрин). – Новосибирск : НГАСУ (Сибстрин), 2013. – 121 с.
3. Методы вычислений в пакете MathCAD : учеб. пособие / И. А. Бедарев [и др.] ; Новосиб. гос. архитектур.-строит. ун-т. – Новосибирск : НГАСУ (Сибстрин), 2013. – 168 с.
4. Информатика : варианты контрольных работ и метод. указания по их выполнению / Новосиб. гос. архитектур.-строит. ун-т (Сибстрин) ; сост.: Н. П. Кисленко, И. Н. Мухина. – Новосибирск : НГАСУ (Сибстрин), 2014. – 36 с.

Составитель
Ирина Николаевна Мухина

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
(Лабораторный практикум в Mathcad)
Лабораторные работы
и методические указания по их выполнению
для студентов, обучающихся
по направлению подготовки 08.03.01 «Строительство»
и специальности 08.05.01 «Строительство
уникальных зданий и сооружений»
всех форм обучения

Редактор А.В. Тренина

Новосибирский государственный архитектурно-строительный
университет (Сибстрин)
630008, Новосибирск, ул. Ленинградская, 113
