



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУ «Информатика и системы управления»

КАФЕДРА ИУ-1 «Системы автоматического управления»

ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ

Студент Шевченко Алексей Дмитриевич
фамилия, имя, отчество

Группа ИУ1-32Б

Тип практики Учебно-технологический практикум на ЭВМ

Название предприятия Кафедра «Системы автоматического управления»

Студент 29/11/2022
(Подпись, дата) А.Д. Шевченко
(И.О. Фамилия)

Руководитель практики 29/11/2022
(Подпись, дата) И.Ю. Ермаков
(И.О. Фамилия)

Оценка _____

2022 г.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э.
Баумана
(национальный исследовательский университет)
(МГТУ им. Н.Э. Баумана)

УТВЕРЖДАЮ

Заведующий кафедрой ИУ-1
(Индекс)

К.А. Неусыпин
(И.О. Фамилия)

« 24 » сентября 20 22 г.

З А Д А Н И Е
на прохождение учебной практики

Студент группы ИУ1-32Б
Шевченко Алексей Дмитриевич
(Фамилия, имя, отчество)

Задание Ознакомление с основами синтаксиса MathWorks MatLAB.
Реализация программного кода. Составление отчёта по пройденной
практике.

Оформление отчета по практике:

Отчет на 28 листах формата А4.

Перечень графического (илюстративного) материала (слайды, фотографии,
скриншоты экрана и т.п.)

Дата выдачи задания « 1 » ноября 20 22 г.

Руководитель И.Ю. Ермаков
Практики 29/11/2022
(Подпись, дата) (И.О. Фамилия)

Студент 29/11/2022 А.Д. Шевченко
(Подпись, дата) (И.О. Фамилия)

Оглавление

.....	0
ВВЕДЕНИЕ.....	3
Тема 1. Начало работы в среде MATLAB	5
1.1. Общие сведения о среде программирования MATLAB	5
1.2. Интерфейс среды MATLAB	6
Тема 2. ПРОГРАММИРОВАНИЕ MATLAB.....	8
2.1. М-файлы.....	8
2.1.1. Сохранение данных в файл	10
2.2. Создание.....	12
2.3. Действительные числа в MATLAB	13
2.3.1. Ввод действительных чисел.....	13
2.3.2. Арифметические операции и встроенные функции.....	14
2.4. Векторы и матрицы в MATLAB	15
2.4.1. Формирование векторов и матриц.....	15
2.4.2. Операции над векторами и матрицами	19
Тема 3.Графики.....	21
3.1. Создание графика	21
3.2. Создание нескольких графиков на одной канве	26
3.3. Построение 3D графиков и поверхностей	29
Практика.....	32
Практика 1	33
Задание 1	33
Задание 2	34
Задание 3	35
Задание 5	41
Задание 6	43
Практика 2	44
Задание 7	44
Задание 8	50
Практика 3	53
Задание 9	53
Задание 10	54
Задание 11	56
Задание 12	57
Задание 13	58
Задание 14	60
ЗАКЛЮЧЕНИЕ	61
Список использованной литературы	62

ВВЕДЕНИЕ

За многие годы накоплены обширные библиотеки научных подпрограмм на различных алгоритмических языках, предназначенных для решения типовых задач вычислительной математики. Кроме того, имеется целый ряд различных математических пакетов, реализующих разнообразные численные методы и производящих аналитические математические преобразования. Наиболее известными сегодня являются пакеты прикладных программ (ППП) и математические библиотеки: MatLab (фирма The MathWorks), Maple (фирма Waterloo Maple Inc), Mathematica (фирма Wolfram Research), MathCAD (фирма MathSoft Inc).



MatLab – MATrix LABoratory (матричная лаборатория). Эта система предназначена для осуществления любых численных расчетов и моделирования технических и физических систем, а также выполнения научных и инженерных расчетов при работе с массивами данных;

MathCAD - является интегрированной системой программирования, ориентированной на проведение математических и инженерно-технических расчетов.

Maple — программный пакет, система компьютерной алгебры. Система Maple предназначена для символьных вычислений, хотя имеет ряд средств и для численного решения дифференциальных уравнений и нахождения интегралов. Обладает развитыми графическими средствами. Имеет собственный язык программирования, напоминающий Паскаль.

Mathematica — система компьютерной алгебры компании Wolfram Research. Содержит множество функций, как для аналитических преобразований, так и для численных расчётов. Кроме того, программа поддерживает работу с графикой и звуком, включая построение двух- и трёхмерных графиков функций, рисование произвольных геометрических фигур, импорт и экспорт изображений и звука.

Цель работы – ознакомление с основами работы со средой MATLAB, изучение базовых функций. Как для дальнейшего углублённого изучения с целью повышения навыков владения системой, так и в качестве вспомогательного помощника в освоении основ рекомендуется ознакомиться встроенной документацией (Documentation).

Тема 1. Начало работы в среде MATLAB

1.1. Общие сведения о среде программирования MATLAB

Система использует математический сопроцессор и допускает обращения к программам, написанным на языках Fortran, С и С++.

Наиболее известные области применения системы MATLAB:

- математика и вычисления;
- разработка алгоритмов;
- вычислительный эксперимент, имитационное моделирование;
- анализ данных, исследование и визуализация результата
- научная и инженерная графика разработка приложений, включая графический интерфейс пользователя.

MATLAB – это интерактивная система, основным объектом которой является массив, для которого не требуется указывать размерность явно. Это позволяет решать многие вычислительные задачи, связанные с векторно-матричными формулами, существенно сокращая время, необходимое для программирования на скалярных языках типа Fortran или С. Будучи ориентированной на работу с реальными данными, эта система выполняет все вычисления в арифметике с плавающей точкой.

1.2. Интерфейс среды MATLAB

После запуска MATLAB на экране появляется основное окно системы MATLAB (рис.1). Рабочий стол системы MATLAB содержит следующие инструментальные окна:

- Command Window – это область ввода команд и выражений на языке программирования среды MATLAB, и где система размещает свои ответы на пользовательские команды, описанные также в М-файлах. При каждом сеансе работы среда MATLAB формирует рабочее пространство, т.е. множество переменных, создаваемых пользователем. Команды вводятся после знака “`>>`”;
- Workspace Browser – отображает переменные, с которыми пользователь работает в данный момент, а также основную информацию о них – размер и тип переменной;
- Editor/Debugger – создание, редактирование и отладка М-файлов;
- Current Folder – предназначено для просмотра файловой структуры и установки текущего.

Команды управления окном командного режима:

- `clc` — очищает экран и размещает курсор в левом верхнем углу пустого экрана.
- `clear all` — удаление элементов из рабочей области, удаление данных из переменных.
- `clear variables` — удаление переменных из рабочей области.
- `close all force` — закрывает все фигуры.

Редактор/отладчик М-файлов выполняет синтаксический контроль программного кода по мере ввода текста. При этом используются следующие цветовые выделения:

- ключевые слова языка программирования — синий цвет;
- операторы, константы и переменные — черный цвет;
- комментарии после знака **%** — зеленый цвет;
- символьные переменные (в апострофах) — пурпурный цвет;
- синтаксические ошибки — красный цвет.

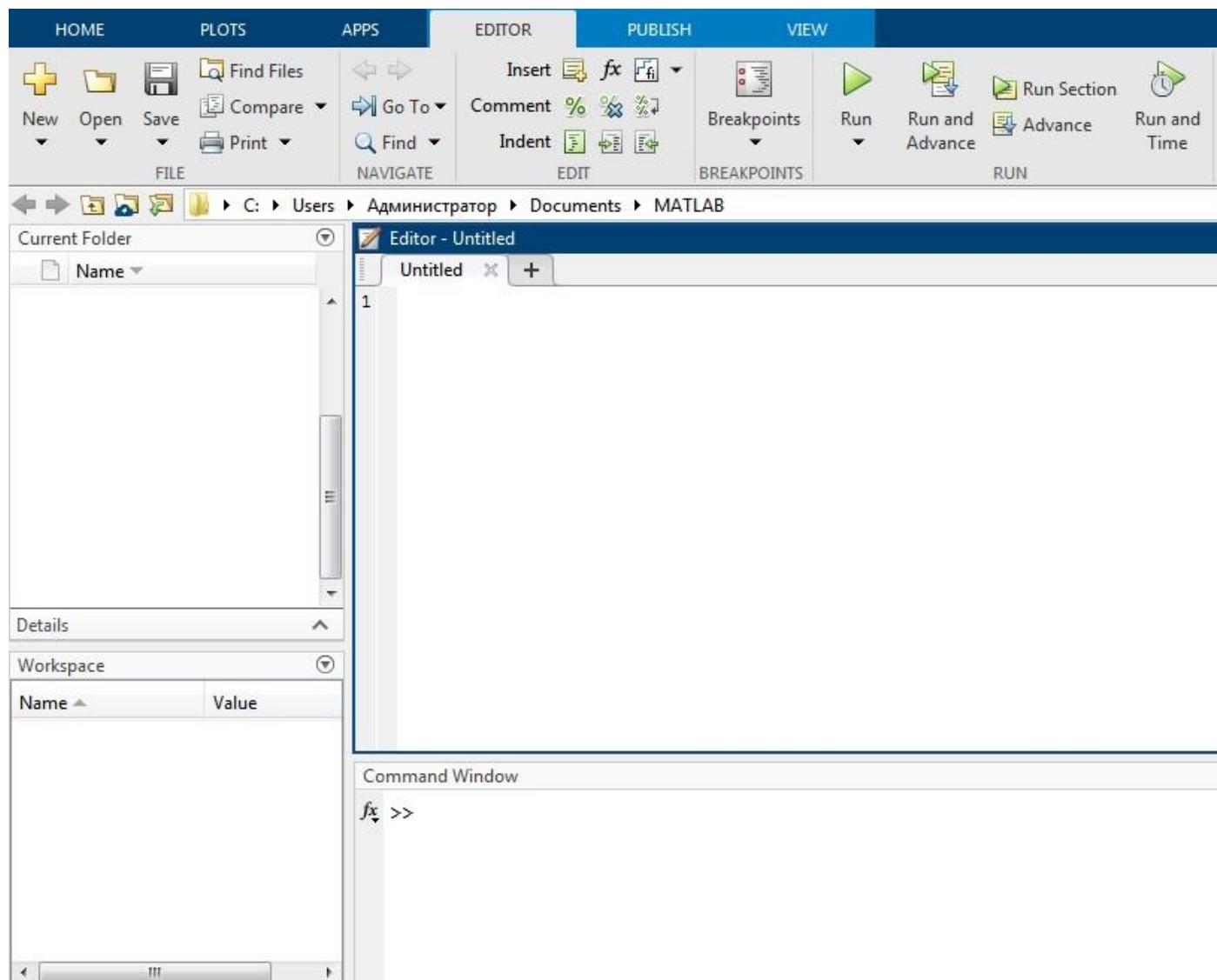


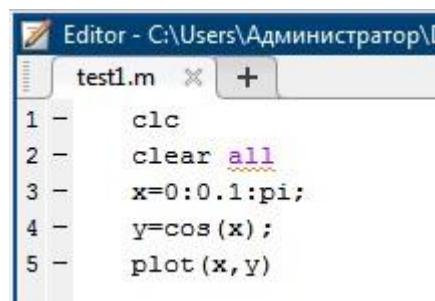
Рис. 1

Тема 2. ПРОГРАММИРОВАНИЕ MATLAB

2.1. M-файлы

В среде программирования MATLAB существует возможность создавать программы 2 типов: скриптовые файл-программы (управляющие программы - Script-файлы) и файлфункции (процедуры).

При помощи скриптовых файлов оформляются основные программы, управляющие от начала до конца организацией вычислительного процесса. Скриптовые файл-программы не имеют входных и выходных аргументов, все переменные, объявленные в файле, становятся доступными в рабочей среде сразу после выполнения. В скриптовых файлах программах все используемые переменные образуют «рабочее пространство»; их значения сохраняются в течение всего сеанса работы с системой. Например:



```

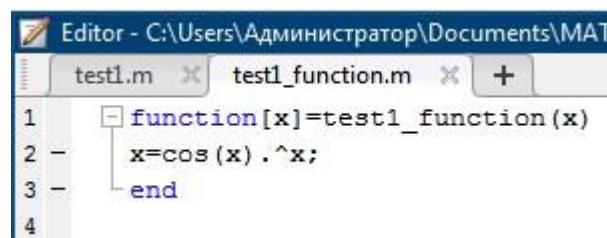
Editor - C:\Users\Администратор\Документы\matlab\work\test1.m
test1.m + 
1 - clc
2 - clear all
3 - x=0:0.1:pi;
4 - y=cos(x);
5 - plot(x,y)

```

Рис. 2

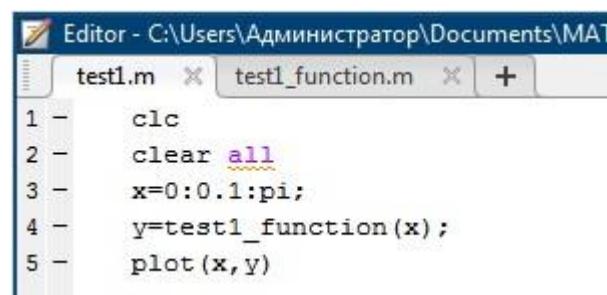
В файл-функции оформляются отдельные процедуры и разработанные пользователем специальные функции (т.е. такие части программы, которые рассчитаны на неоднократное использование при изменяемых входных параметрах). Функции могут быть полезны при программировании собственных приложений, функции производят необходимые действия с входными аргументами, согласно составленной пользователем последовательности операций, и возвращают результат в выходных аргументах.

В файлфункциях все имена переменных внутри файла, а также имена переменных, указанные в заголовке, воспринимаются как локальные, т. е. все значения этих переменных после завершения работы процедуры исчезают, и область оперативной памяти компьютера, которая была отведена под запись значений этих переменных, освобождается для записи в нее значений других переменных. Например:



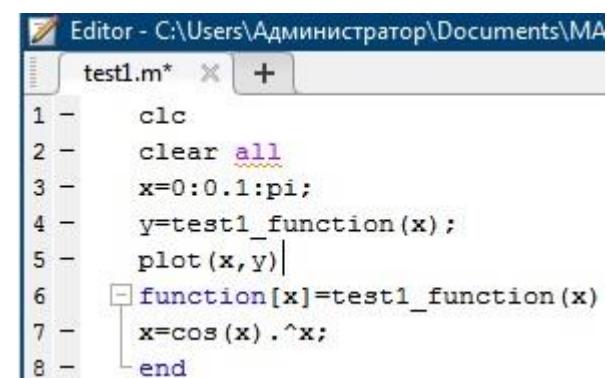
```
Editor - C:\Users\Администратор\Documents\MATLAB
test1.m x test1_function.m x +
1 function[x]=test1_function(x)
2 x=cos(x).^x;
3 end
4
```

Рис. 3



```
Editor - C:\Users\Администратор\Documents\MATLAB
test1.m x test1_function.m x +
1 clc
2 clear all
3 x=0:0.1:pi;
4 y=test1_function(x);
5 plot(x,y)
```

Рис. 4 (Использование данной функции в test1.m)



```
Editor - C:\Users\Администратор\Documents\MATLAB
test1.m* x +
1 clc
2 clear all
3 x=0:0.1:pi;
4 y=test1_function(x);
5 plot(x,y)
6 function[x]=test1_function(x)
7 x=cos(x).^x;
8 end
```

Рис. 5 (Встроенная функция)

Отдельно следует упомянуть об анонимных функциях - функции, которые НЕ имеют заданного имени, а по сути своей являются указателем на функцию, могут принять несколько входных параметров и возвратить один выходной параметр (рис.6)

```
Editor - C:\Users\Администратор
test1.m + 
1 - clear all
2 - clc
3 - a=@(x,y) x+y^x;
4 - disp(a(3,1))|
```

Command Window
4

Рис.6

2.1.1. Сохранение данных в файл

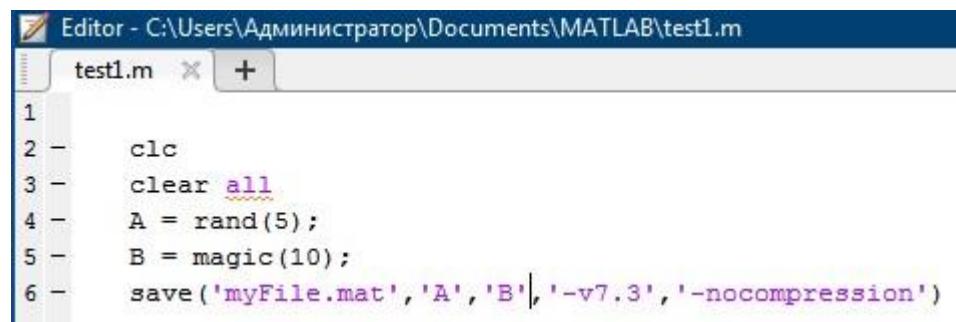
MATLAB позволяет сохранять значения переменных в виде бинарных файлов с расширением .mat. Для этого служит команда save (рис.7), которая может использоваться в ряде форм:

- save fname – записывается рабочая область всех переменных в файле бинарного формата с именем fname.mat;
- save fname X – записывает только значение переменной X;
- save fname X Y Z – записывает значения переменных X, Y и Z.

После параметров команды save можно указать ключи, уточняющие формат записи файлов:

- -mat – двоичный МАТ формат, используемый по умолчанию;
- -ascii – ASCII формат единичной точности (8 цифр);
- -ascii -double – ASCII формат двойной точности (16 цифр);
- -ascii -double -tabs – формат с разделителем и метками табуляции;
- -append – добавление в существующий МАТ файл.

Возможно использование слова save и в формате функции, а не команды, например: `save('fname','var1','var2')` В этом случае имена файлов и переменных задаются строковыми константами.



The screenshot shows the MATLAB Editor window with the title bar "Editor - C:\Users\Администратор\Documents\MATLAB\test1.m". The editor displays the following MATLAB script:

```
1
2 -    clc
3 -    clear all
4 -    A = rand(5);
5 -    B = magic(10);
6 -    save('myFile.mat', 'A', 'B', '-v7.3', '-nocompression')
```

Рис. 7

2.2. Создание

Повторюсь, что любую последовательность команд в MATLAB можно оформить в виде скрипта-файла. Для создания и редактирования таких файлов служит специальный редактор М-файла. Его пустое окно открывается командой **New** (Новый файл), которую можно ввести из позиции **Home** или **Editor** меню окна MATLAB.

Для исполнения необходимо выполнить одно из приведенных действий:

- набрать последовательность команд или имя файла (без .m) в Command Window и нажать кнопку Enter ;
- нажать кнопку F5 на клавиатуре;
- вызвать команду **Editor->Run** из меню редактора М-файлов.

Сохранение файла производится по пути **Editor->Save** или при исполнении:

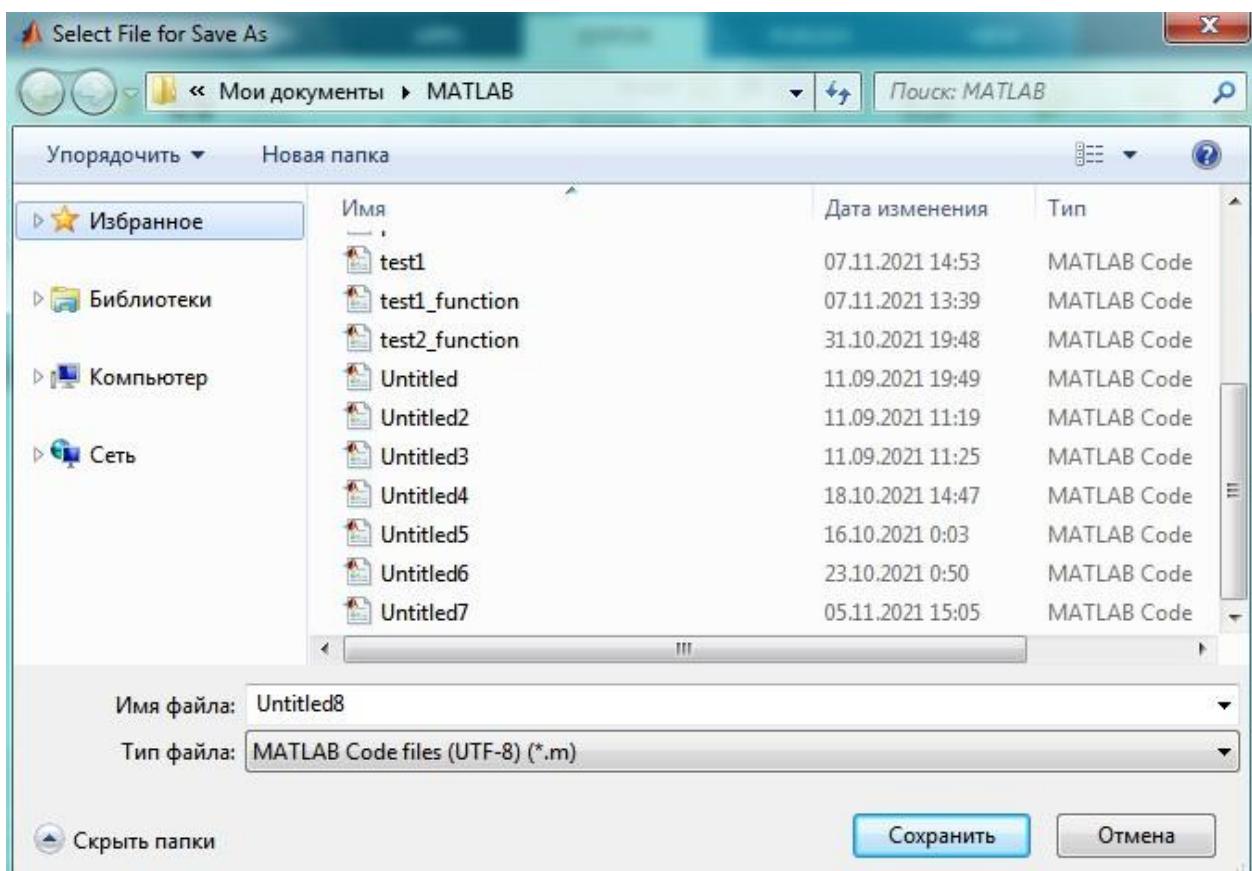


Рис. 8

2.3. Действительные числа в MATLAB

2.3.1. Ввод действительных чисел

Ввод чисел с клавиатуры производится по общим правилам, принятым для языков программирования высокого уровня:

1. для отделения дробной части числа применяется десятичная точка (вместо запятой при обычной записи);
2. десятичный показатель числа записывается в виде целого числа после предварительной записи символа **e** (пример: **1.2305e-5**).

Вывод промежуточной информации в командное окно подчиняется следующим правилам:

1. если запись оператора не заканчивается символом ";", результат действия выводится в командное окно;
2. если оператор заканчивается символом ";", результат его действия не отображается в командном окне. В таком случае используют команду **disp;**
3. если оператор не содержит знака присваивания (=), т.е. является просто записью некоторой последовательности действий над числами и переменными, то значение результата присваивается специальной системной переменной **ans;**

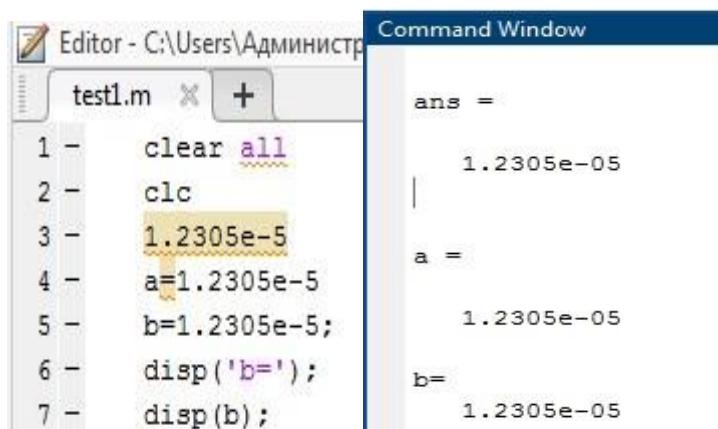


Рис. 9

2.3.2. Арифметические операции и встроенные функции

В арифметических выражениях языка MATLAB используются следующие знаки арифметических операций:

- Сложение – «+»;
- Вычитание – «-»;
- Умножение – «*»;
- Возвведение в степень – «^»;
- Деление чисел – «/»;

MATLAB содержит в себе все распространенные математические функции, которые доступны по их имени при реализации алгоритмов.

- `sqrt(x)` - вычисление квадратного корня
- `exp(x)` - возвведение в степень числа е
- `pow2(x)` - возвведение в степень числа 2
- `log(x)` - вычисление натурального логарифма
- `log10(x)` - вычисление десятичного логарифма
- `log2(x)` - вычисление логарифма по основанию 2
- `sin(x)` - синус угла x, заданного в радианах
- `cos(x)` - косинус угла x, заданного в радианах
- `tan(x)` - тангенс угла x, заданного в радианах
- `cot(x)` - котангенс угла x, заданного в радианах
- `asin(x)` - арксинус
- `acos(x)` - арккосинус
- `atan(x)` - арктангенс
- `pi` - число - пи
- `round(x)` - округление до ближайшего целого
- `fix(x)` - усечение дробной части числа
- `floor(x)` - округление до меньшего целого
- `ceil(x)` - округление до большего целого

- `mod(x)` - остаток от деления с учётом знака
- `sign(x)` - знак числа
- `factor(x)` - разложение числа на простые множители
- `isprime(x)` - истинно, если число простое
- `rand` - генерация псевдослучайного числа с равномерным законом распределения
- `randn` - генерация псевдослучайного числа с нормальным законом распределения
- `abs(x)` - вычисление модуля числа

2.4. Векторы и матрицы в MATLAB

Описанные выше простые правила вычислений распространяются и на гораздо более сложные вычисления, которые (при использовании обычных языков программирования) требуют составления специальных программ. MATLAB является системой компьютерных вычислений, специально предназначеннной для осуществления сложных операций с векторами, матрицами и полиномами. Все данные в системе MATLAB представлены в виде различных массивов – векторов и матриц. Под вектором в MATLAB понимается одномерный массив чисел, а под матрицей – двумерный массив.

2.4.1. Формирование векторов и матриц

Исходные значения вектора-строки можно задавать путем поэлементного ввода. Для этого вначале указывают имя вектора, затем ставят знак присваивания “=”, далее открывающую квадратную скобку “[“, за ней значения вектора, отделяя их между собой пробелами или запятыми. Запись завершается закрывающей квадратной скобкой “]”. Вектор-столбец задается аналогично вектору строке, но элементы отделяются друг от друга знаком “;”. Для создания вектора-столбца можно также использовать форму записи вектора-строки с указанием

значений через пробел, при этом в конце добавляется апостроф ' (рис.10).

The screenshot shows the MATLAB Editor window with a script named 'test1.m' containing the following code:

```

1 - clear all
2 - clc
3 - A=[1 2 3 4]
4 - A=[1 2 3 4] '

```

The Command Window below displays the output of the script:

```

A =
1     2     3     4

A =
1
2
3
4

```

Рис. 10

Существует и сокращенный метод ввода вектора, элементы которого являются арифметической прогрессией (рис.11):

A = nz:h:kz где: **nz** – начальное значение прогрессии (первый элемент вектора), **kz** – конечное значение прогрессии (последний элемент вектора), **h** – разность прогрессии (шаг).

The screenshot shows the MATLAB Editor window with a script named 'test1.m' containing the following code:

```

1 - clear all
2 - clc
3 - A=1:0.2:1.4
4

```

The Command Window below displays the output of the script:

```

A =
1.0000    1.2000    1.4000

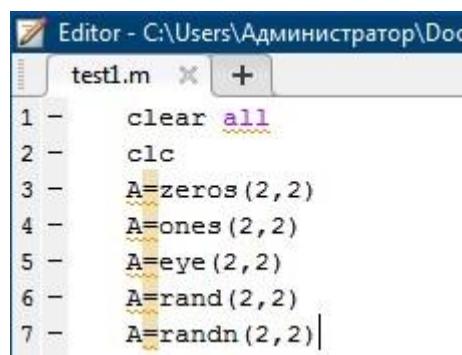
```

Рис. 11

Ввод элементов матрицы осуществляется по строкам. При этом элементы строки матрицы отделяются друг от друга пробелами или запятыми, а строки отделяются друг от друга знаком «;».

Недостатком описанного способа задания (инициализации) матриц является сложность определения матриц больших размеров, состоящих, например, из $100*100$ или $1000*1000$ элементов. Чтобы решить данную задачу, в MATLAB существуют функции инициализации матриц нулями, единицами или случайными значениями (рис.12-13):

- **`zeros(M,N)`** - создает матрицу размером $(M*N)$ с нулевыми элементами;
- **`ones(M,N)`** - создает матрицу размером $(M*N)$ с единичными элементами;
- **`eye(M,N)`** - создает матрицу размером $(M*N)$ с единицами по главной диагонали и остальными нулевыми элементами;
- **`rand(M,N)`** - создает матрицу размером $(M*N)$ из случайных чисел, равномерно распределенных в диапазоне от 0 до 1;
- **`randn (M, N)`** – создает матрицу размером M на N из случайных чисел, распределенных по нормальному закону в диапазоне от 0 до 1.



The screenshot shows the MATLAB Editor window titled 'Editor - C:\Users\Администратор\Doc'. It contains a script named 'test1.m' with the following code:

```

1 - clear all
2 - clc
3 - A=zeros(2,2)
4 - A=ones(2,2)
5 - A=eye(2,2)
6 - A=rand(2,2)
7 - A=randn(2,2)

```

Рис. 12

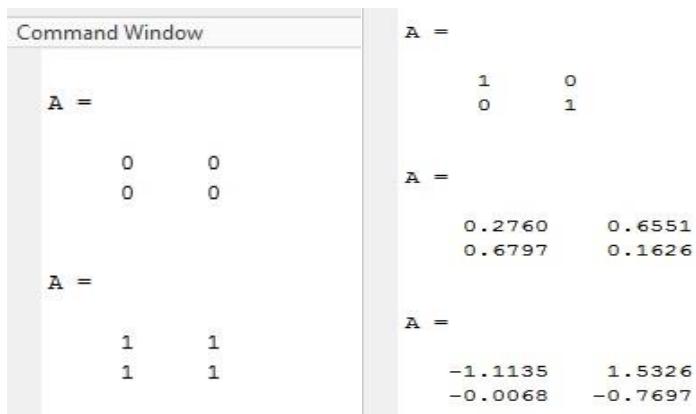


Рис. 13

Для доступа к тому или иному элементу матрицы используется следующие конструкции:

- $A(i, j)$ – обращение к элементу i -й строки j -го столбца;
- $A(i, :)$ – обращение к i -й строке;
- $A(:, j)$ – обращение к j -му столбцу;
- $A(i, j) = []$ – удаление элемента, находящийся в i -й строке и j -м столбце;
- $A(i, j) = 2$ – элементу, находящемуся в i -й строке, j -м столбце,

присваивается значение необходимое значение (здесь - 2).

Последние две операции применяются также ко всей строке или столбцу матрицы. **find()** – возвращает набор индексов неравных нулю элементов матрицы или вектора. **nnz()** - возвращает количество неравных нулю элементов матрицы или вектора.

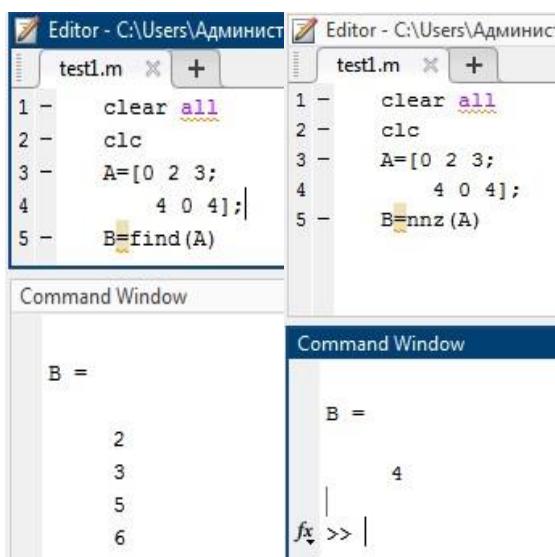


Рис. 14

2.4.2. Операции над векторами и матрицами

Базовые действия с матрицами и векторами - сложение, вычитание, транспонирование, умножение на число, умножение матрицы на матрицу (вектора на вектор), возвведение в целую степень - осуществляются в языке MATLAB с помощью обычных знаков арифметических операций: «+», «-», «*», «^» - соответственно.

При использовании этих операций важно помнить условия, при которых эти операции являются возможными:

- при сложении или вычитании матрицы должны иметь одинаковые размерности - при умножении матриц количество столбцов первой матрицы должно совпадать с количеством строк второй матрицы.

Наряду с операциями над отдельными элементами векторов и матриц система позволяет производить операции умножения, деления и возвведения в степень сразу над всеми элементами. Для этого перед знаком операции ставится точка. Так например, если оператор * означает умножение, то оператор .* - поэлементное умножение. Следуя из этого получаем :.*; ./; .^.

Пример некоторых действий – рисунок 15.

The screenshot shows the MATLAB environment. The Editor window at the top contains the script file 'test1.m' with the following code:

```

Editor - C:\Users\Администратор\Documents
test1.m  ×  +
1 -      clear all
2 -      clc
3 -      A=[1,2,3;4,5,6;7,8,9];
4 -      D=[1,0,1];
5 -      B=A.*D;disp(B);
6 -      C=D*A;disp(C);

```

The Command Window below displays the results of the execution:

```

Command Window
    1      0      3
    4      0      6
    7      0      9
    8     10     12

```

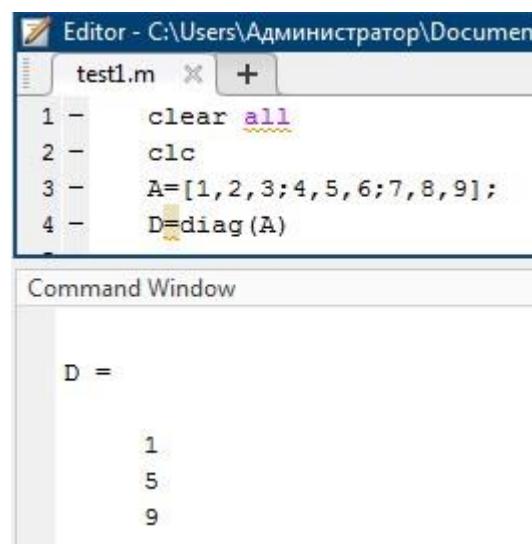
Рис. 15

Команды, реализующие функции определения матриц:

- **inv(A)** – выполняет операцию обращения квадратной матрицы A, определитель которой отличен от нуля;
- **diag(A, k)** – формирует вектор-столбец, содержащий элементы главной или k-й диагонали матрицы A (Рис.16);
- **det(A)** – вычисление определителя матрицы A;
- **rank(A)** – вычисление ранга матрицы A.

Команды, реализующие функции операций над векторами:

- **length(X)** – возвращает длину вектора X;
- **size(A)** – возвращает размер матрицы A.



The screenshot shows the MATLAB environment. The Editor window at the top contains a script named 'test1.m' with the following code:

```

Editor - C:\Users\Администратор\Document
test1.m  ×  +
1 -      clear all
2 -      clc
3 -      A=[1,2,3;4,5,6;7,8,9];
4 -      D=diag(A)

```

The Command Window below shows the output of the code execution:

```

Command Window
D =
    1
    5
    9

```

Рис. 16

Тема 3.Графики

3.1. Создание графика

Графики в MATLAB строятся в отдельных масштабируемых и перемещаемых окнах. Следует помнить, что MATLAB строит графики функций по ряду точек, соединяя их отрезками прямых, то есть осуществляя линейную интерполяцию функции в интервале между смежными точками.

Для построения графика достаточно вначале задать вектор изменения аргумента в формате арифметической прогрессии с начальным и конечным значениями и шагом использовать команду построения графиков **plot** (рис. 17 - 18).

Одна из форм записи: **plot(x1, y1, s1, x2, y2, s2,...)**, где x1, y1 - заданные векторы, элементами которых являются массивы значений аргумента (x1) и функции (y1), отвечающие первой кривой графика; x2, y2 - массивы значений аргумента и функции второй кривой и т.д.

Переменные s1, s2,... являются символьными. Любая из них может содержать до трех специальных символов, определяющих соответственно:

- а) тип линии, которая соединяет отдельные точки графика;
- б) тип точки графика;
- в) цвет линии.

Дополнительно: **grid** - формирует координатную сетку; **xlabel()**, **ylabel()** – подписи соответствующим осям; **title()** - заголовок графика.

Editor - C:\Users\Администратор\Documents\MATLAB

```

test1.m  × + 
1 -      clear all
2 -      clc
3 -      x=0:pi/10:2*pi;
4 -      y=sin(x);
5 -      plot(x,y,'-.r')
6 -      grid
7 -      title('График функции y=sin(x)')
8 -      xlabel('x')
9 -      ylabel('y')

```

Рис. 17

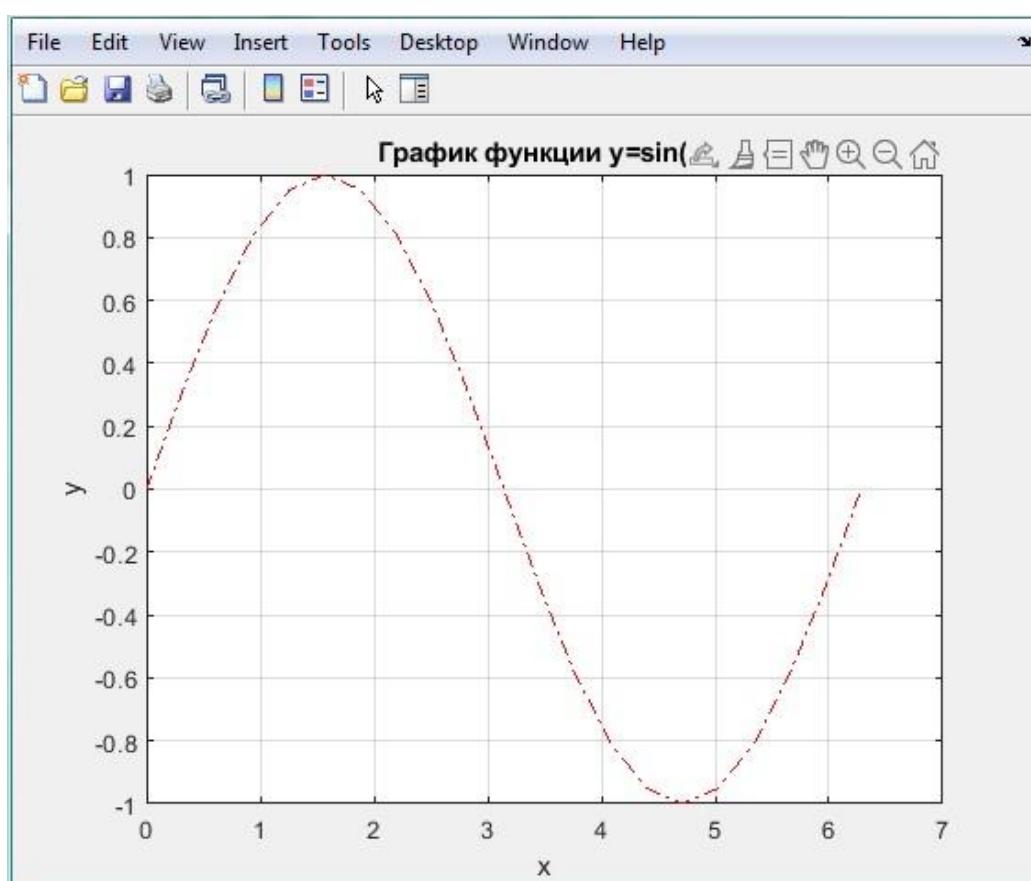
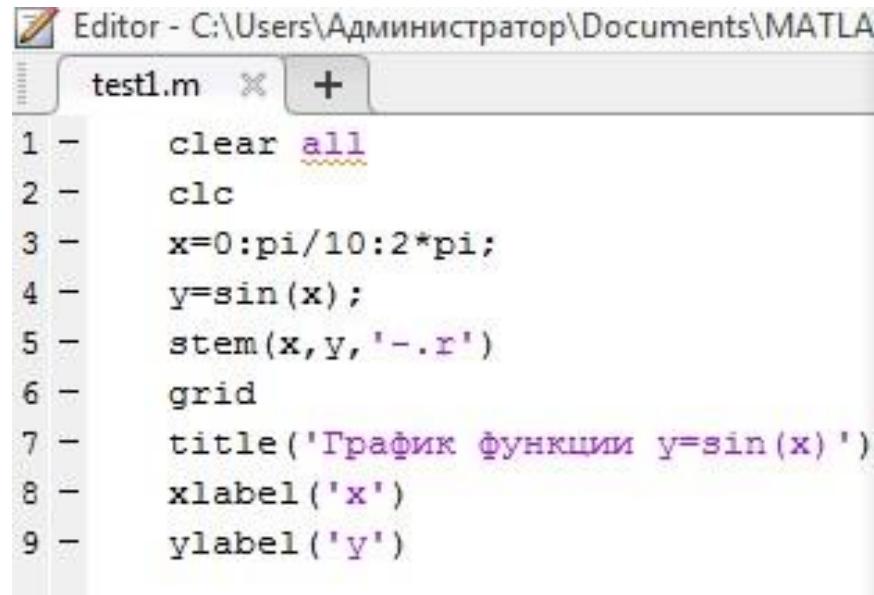


Рис. 18

Иные функции построения графиков

- **stem** (Рис.27-28) – построение дискретных графиков;



Editor - C:\Users\Администратор\Documents\MATLAB

```

1 - clear all
2 - clc
3 - x=0:pi/10:2*pi;
4 - y=sin(x);
5 - stem(x,y,'-.r')
6 - grid
7 - title('График функции y=sin(x)')
8 - xlabel('x')
9 - ylabel('y')

```

Рис. 19

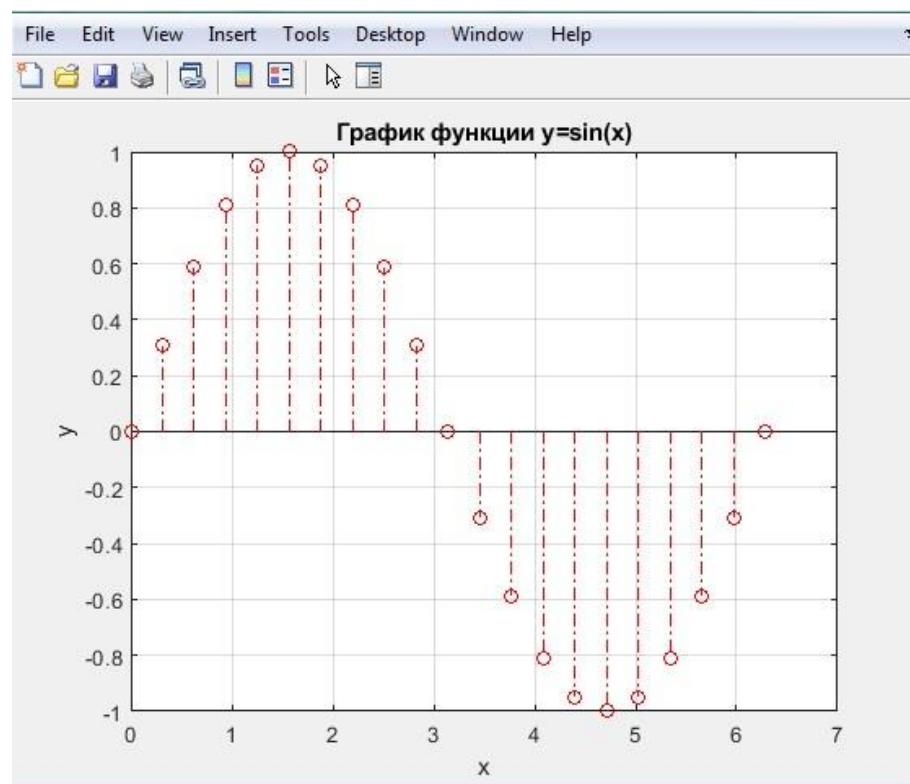
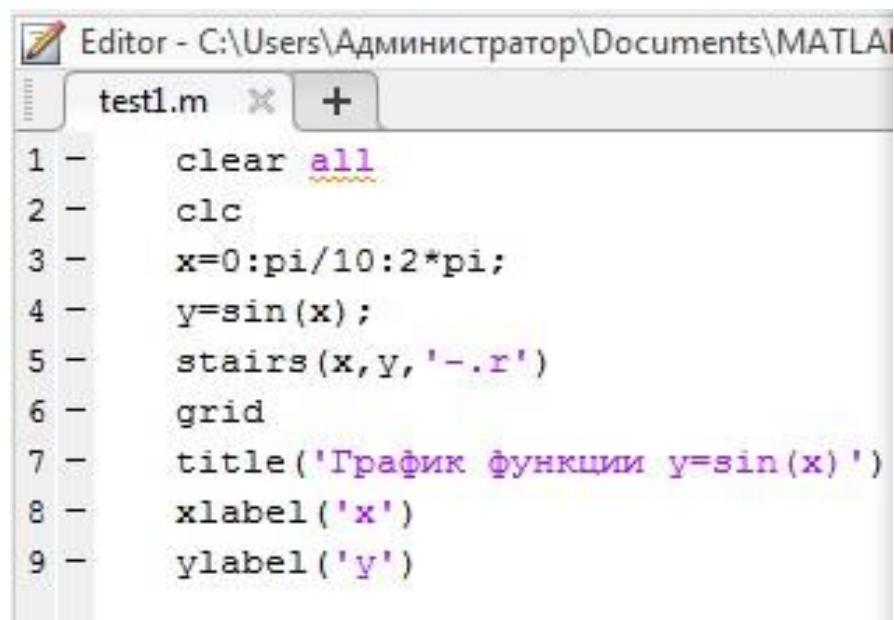


Рис. 20

- **stairs** (Рис. 29-30) – построение ступенчатых графиков;



Editor - C:\Users\Администратор\Documents\MATLAB

```

1 - clear all
2 - clc
3 - x=0:pi/10:2*pi;
4 - y=sin(x);
5 - stairs(x,y,'-.r')
6 - grid
7 - title('График функции y=sin(x)')
8 - xlabel('x')
9 - ylabel('y')

```

Рис. 21

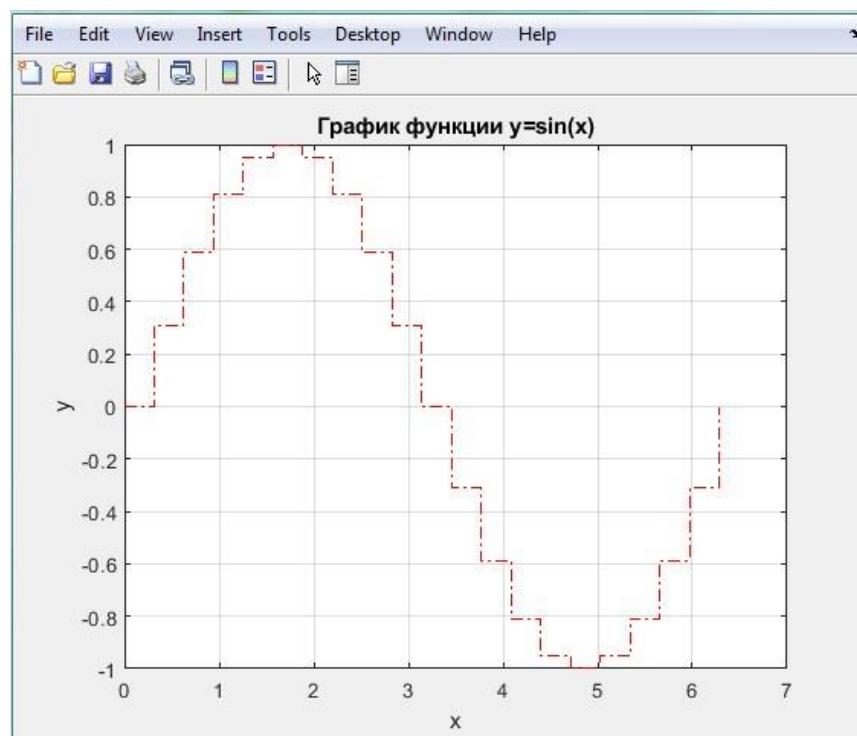
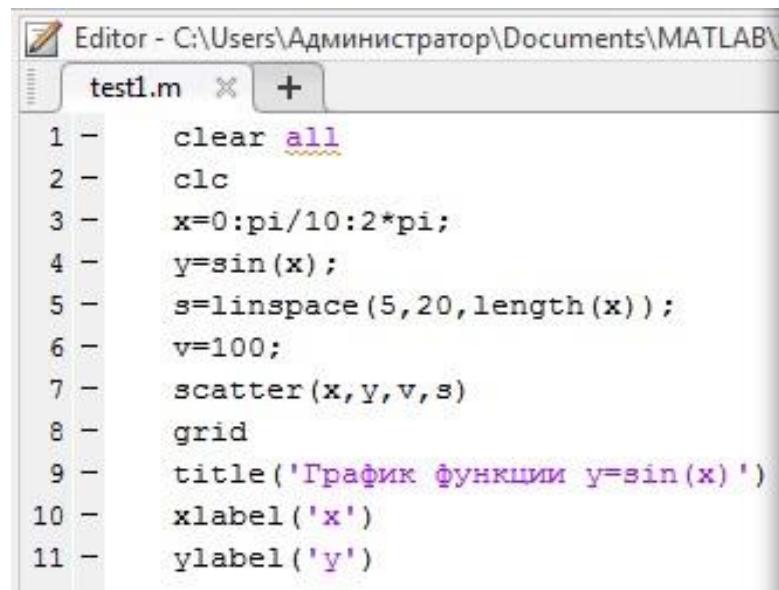


Рис. 22

- **scatter** (Рис.31-32) – точечный график;



```

Editor - C:\Users\Администратор\Documents\MATLAB\
test1.m  × + 
1 -      clear all
2 -      clc
3 -      x=0:pi/10:2*pi;
4 -      y=sin(x);
5 -      s=linspace(5,20,length(x));
6 -      v=100;
7 -      scatter(x,y,v,s)
8 -      grid
9 -      title('График функции y=sin(x)')
10 -     xlabel('x')
11 -     ylabel('y')

```

Рис. 23

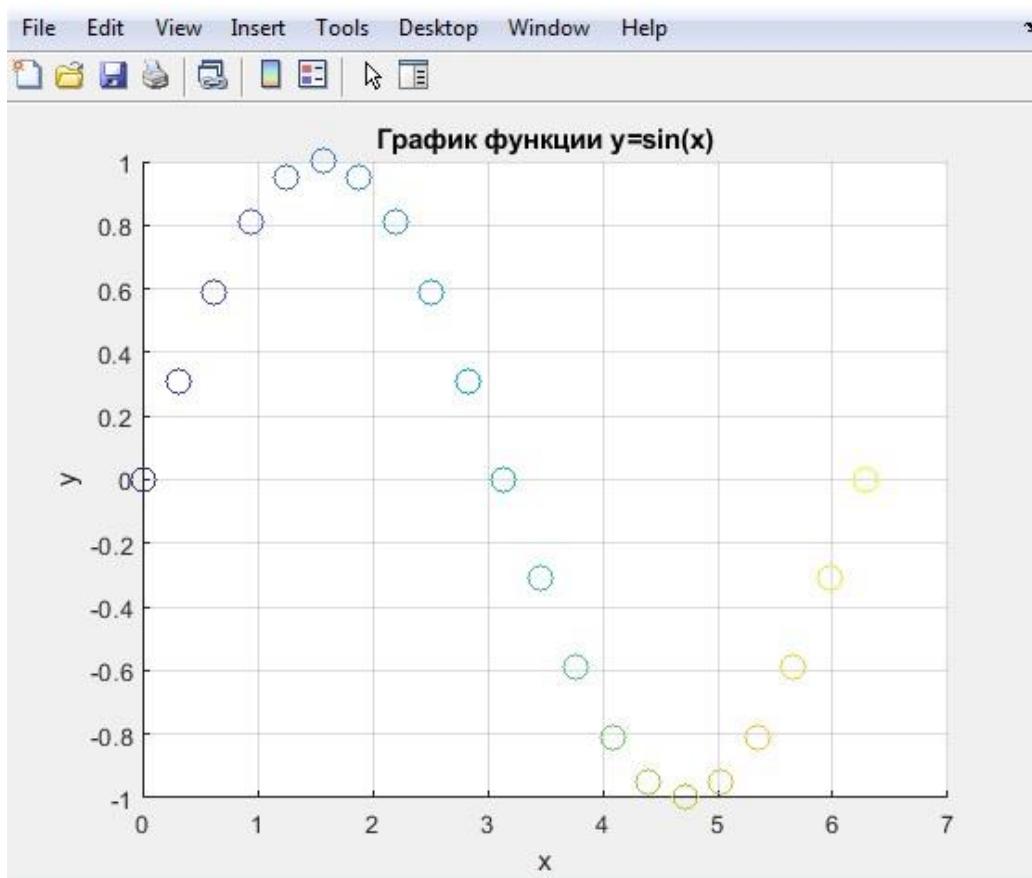


Рис. 24

3.2. Создание нескольких графиков на одной канве

- **yyaxis** (Рис.33) – построение двух кривых с двумя вертикальными осями;

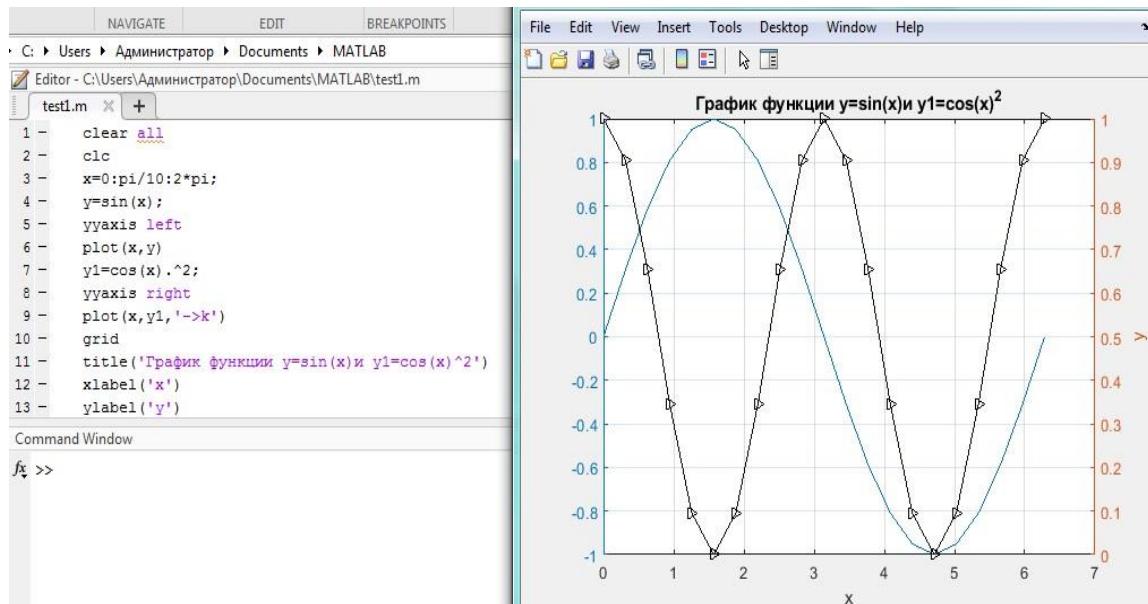


Рис. 25

- **subplot** (Рис.34);

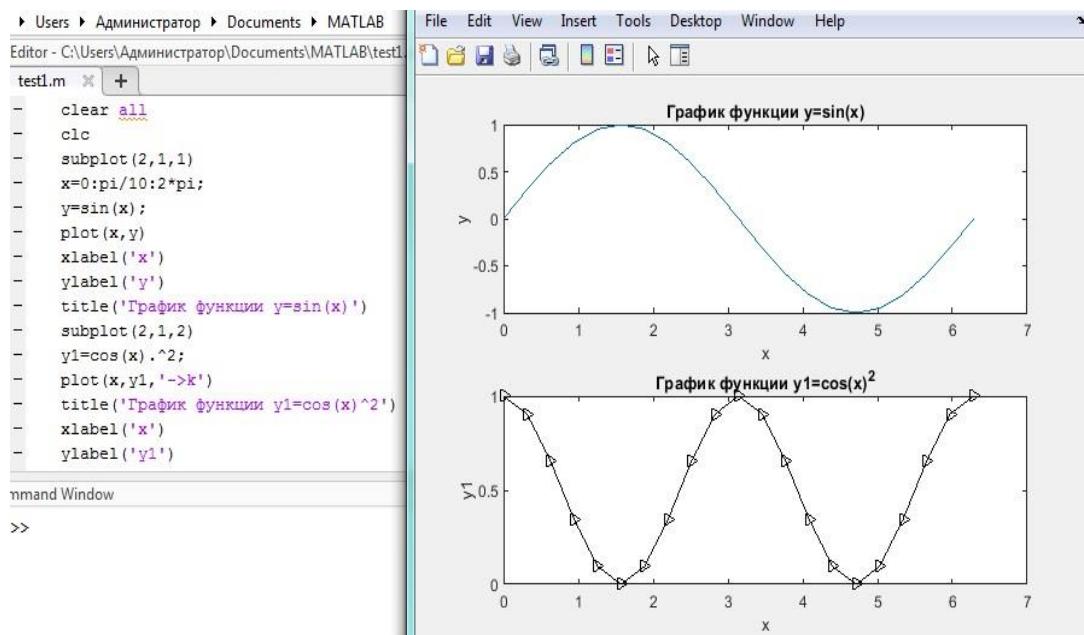


Рис. 26

- **stackedplot** (Рис.35);

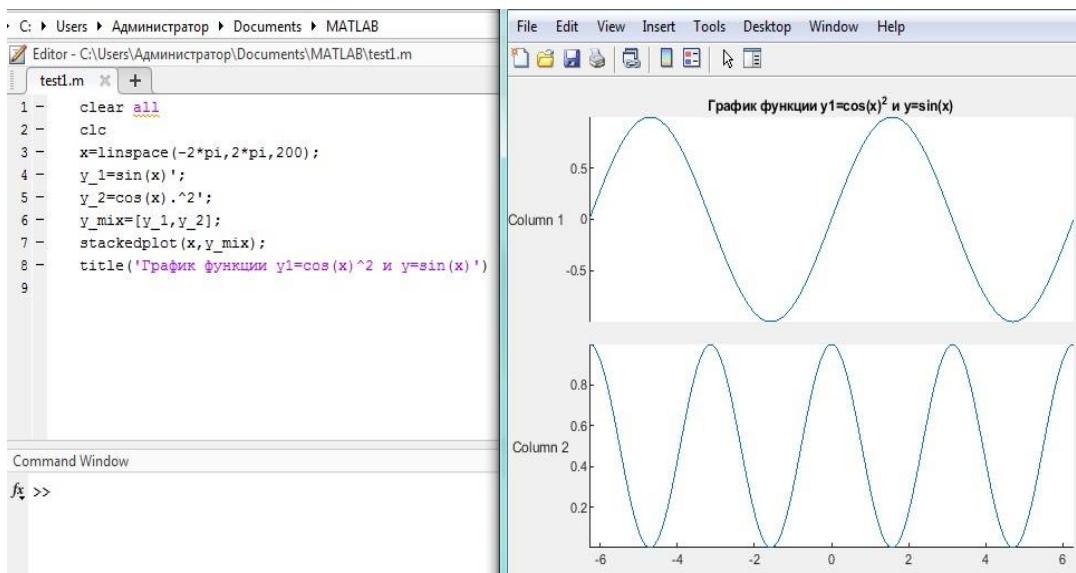


Рис. 27

- **errorbar** (Рис.36);

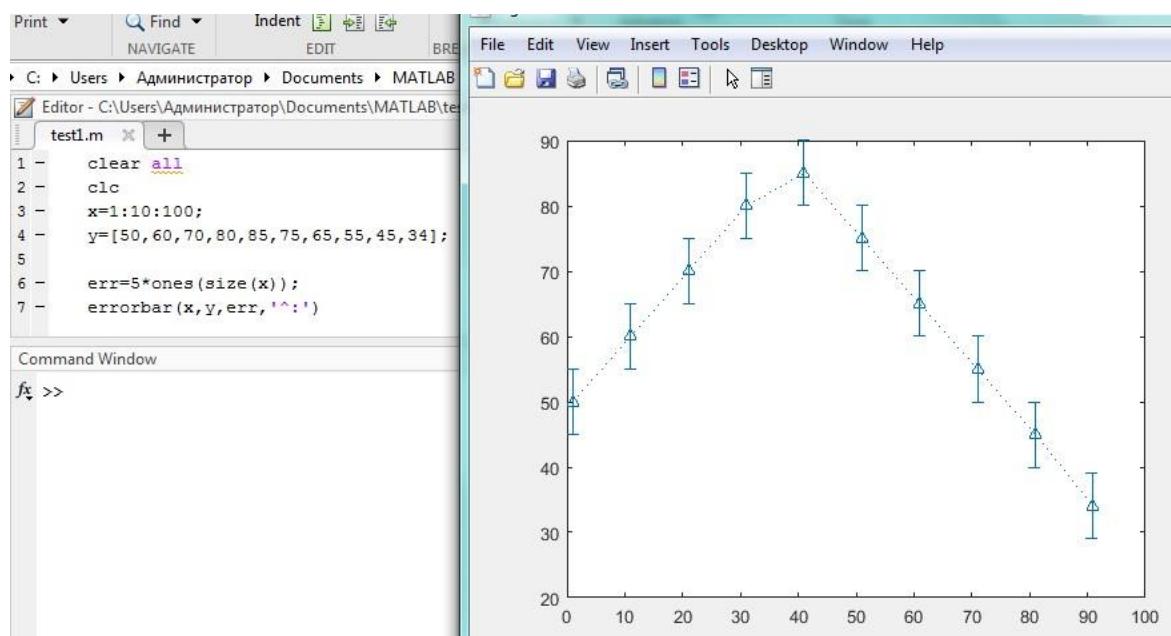


Рис. 28

- **bar** (Рис.37);

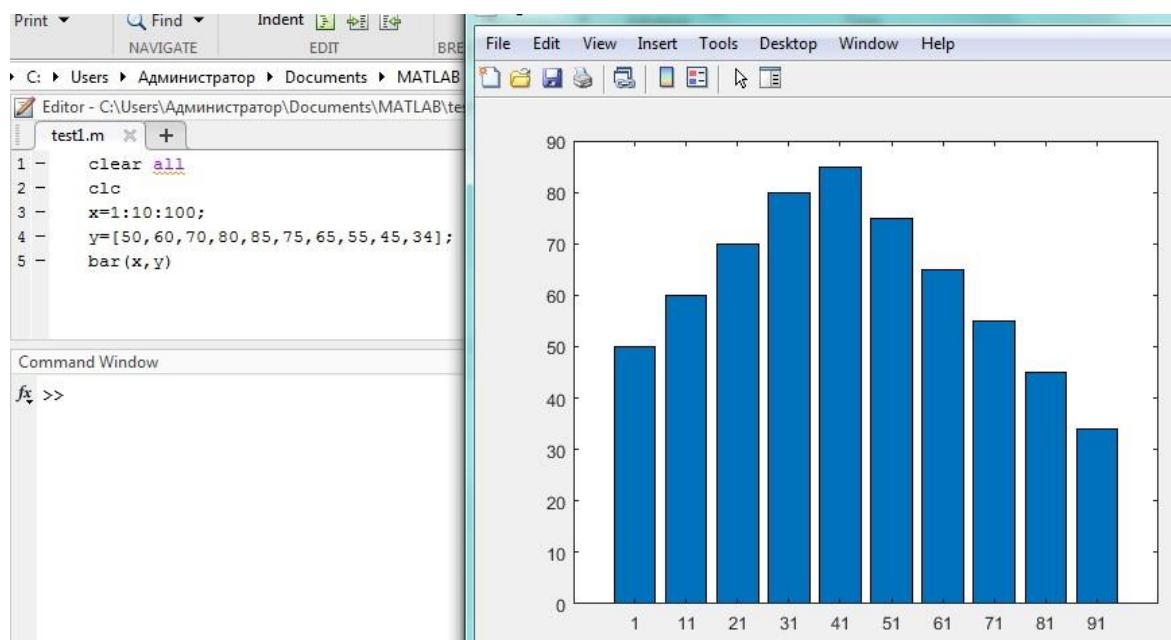


Рис. 29

- **plotmatrix** (Рис.38).

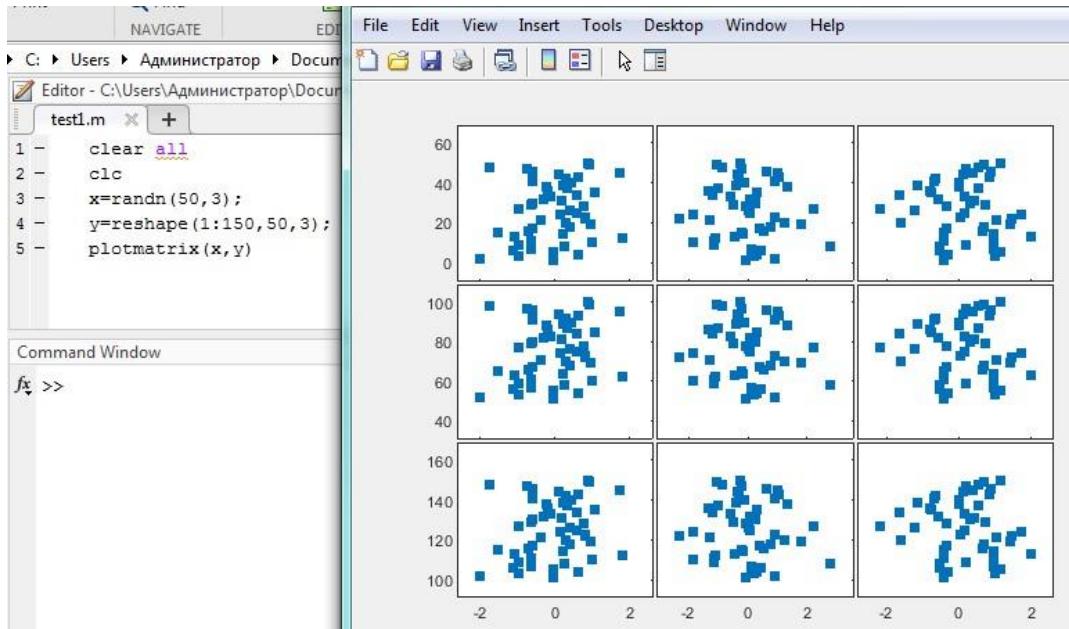


Рис. 30

3.3. Построение 3D графиков и поверхностей

Функции : **plot3** (рис.39),**scatter3** (рис.40),**bar3** (рис.41) – графики; **surf** (рис.42),**mesh** (рис.43) – поверхности.

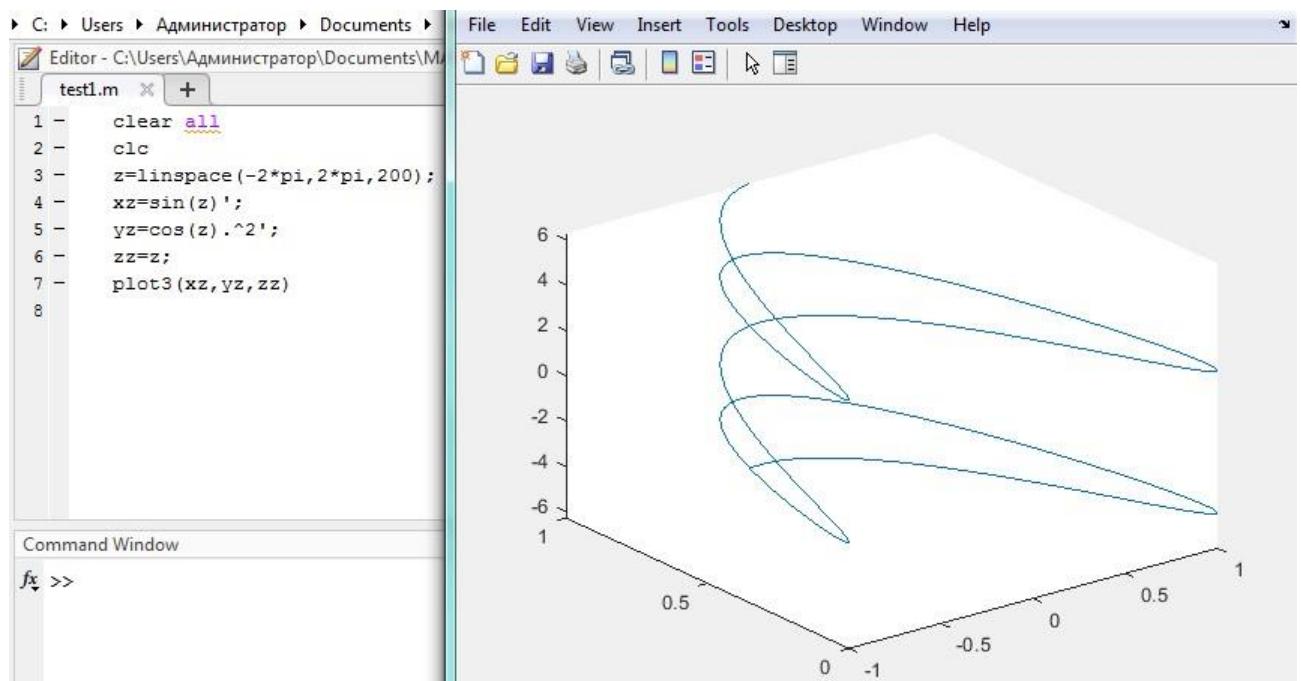


Рис. 31

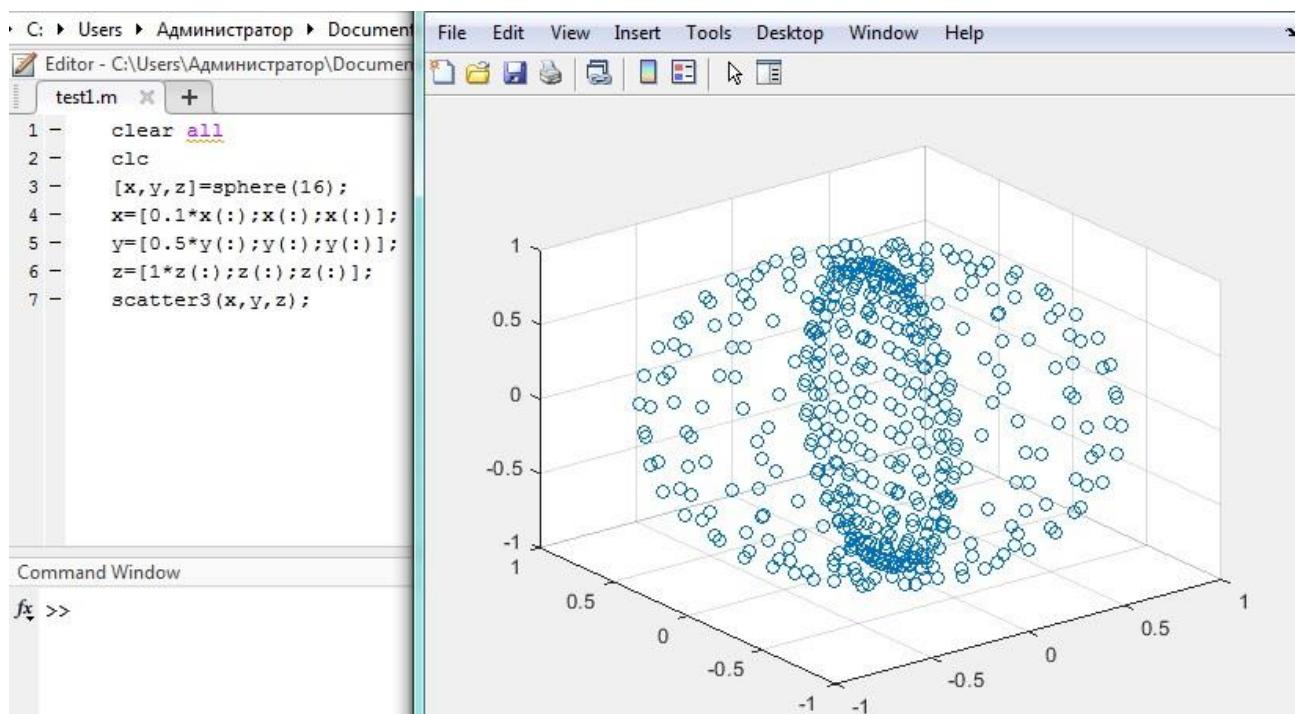


Рис. 32

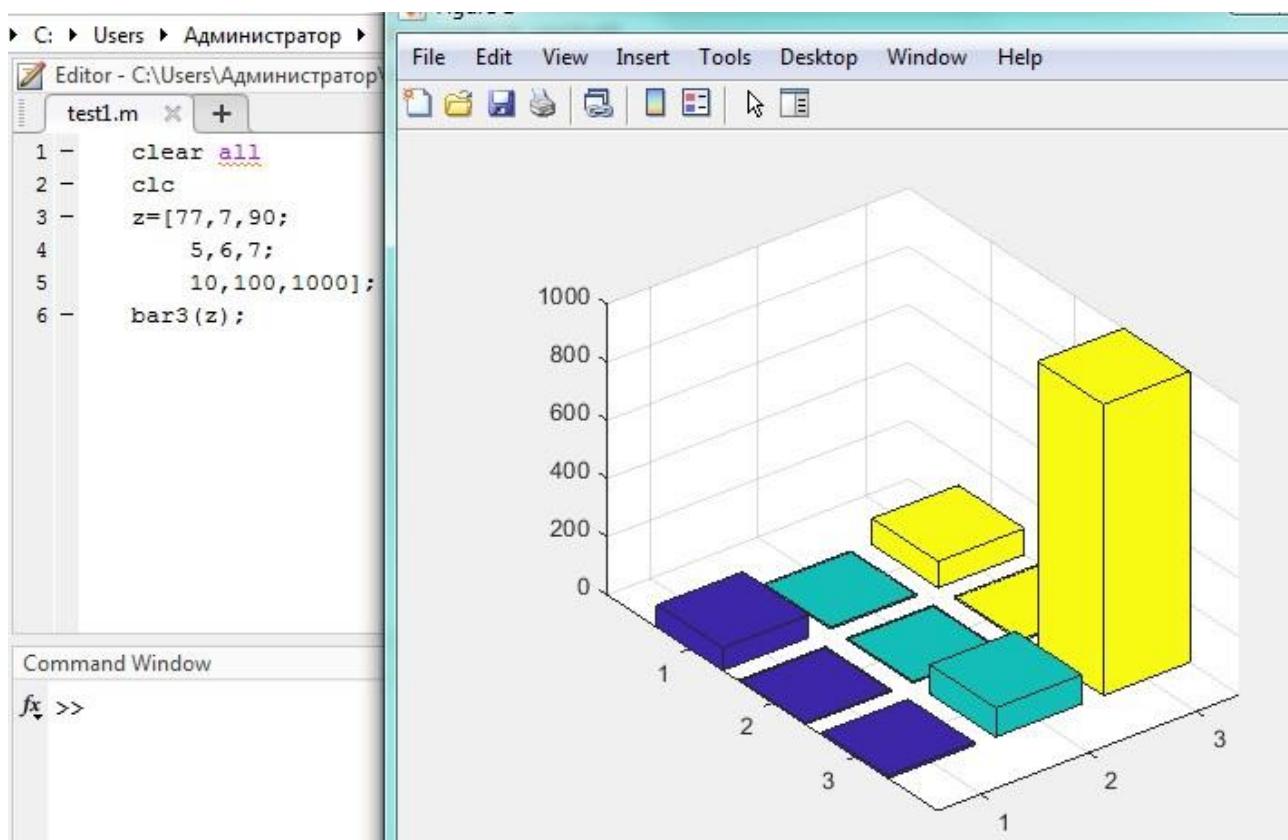


Рис. 33

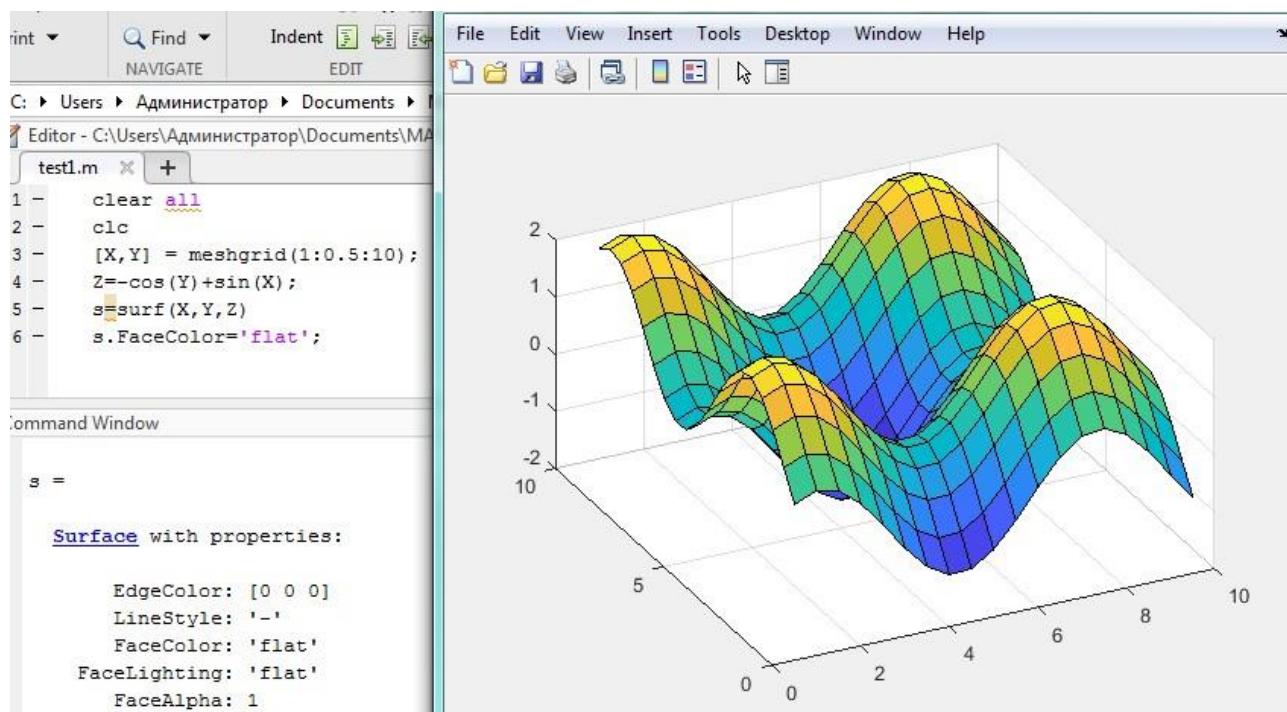


Рис. 34

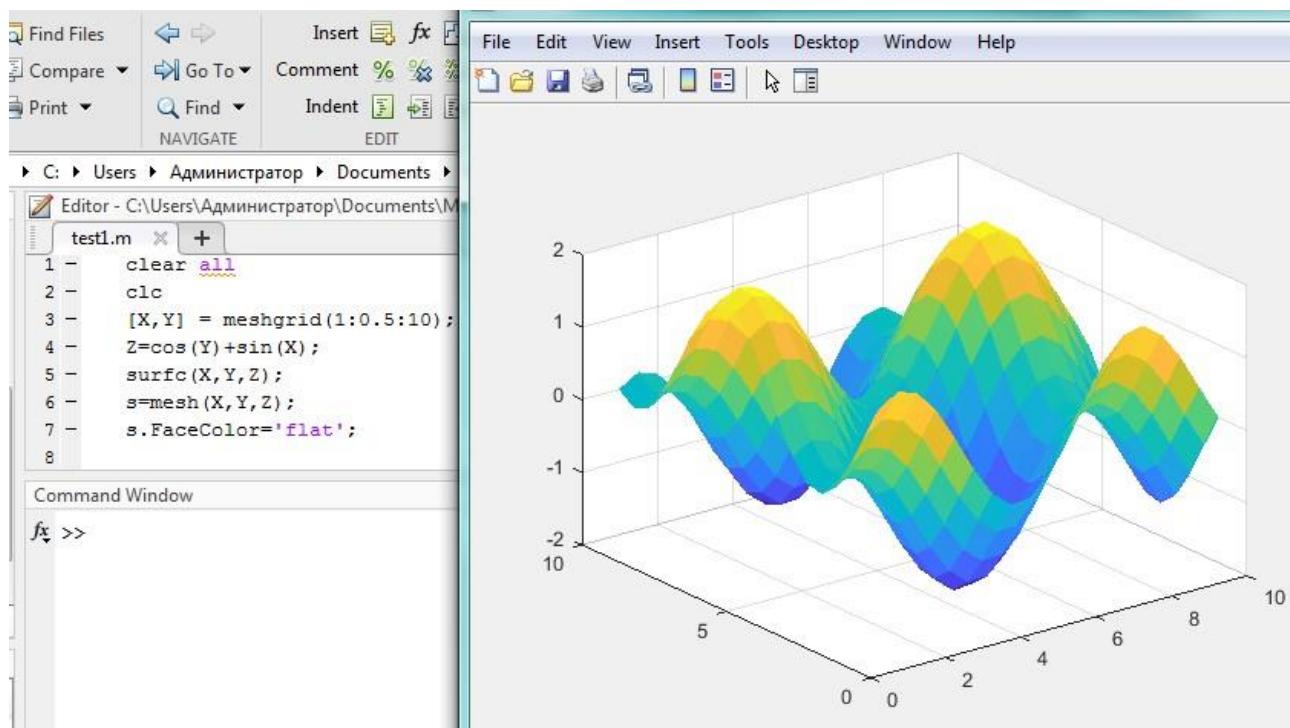


Рис. 35

Практика

Вариант по списку № 25 - Вариант по заданиям № 5

Для хранения кода использовал web-service GitHub

Ссылка на репозиторий:

<https://github.com/BMSTU-Automatic-Control-Systems-IU1-1/project-technological-practice-matlab>

В ходе изучения дисциплины мной были изучены теоретические материалы о работе с MatLAB и выполнены практические задания с практик 1, 2 и 3.

Общая структура файлов представлена на фото ниже

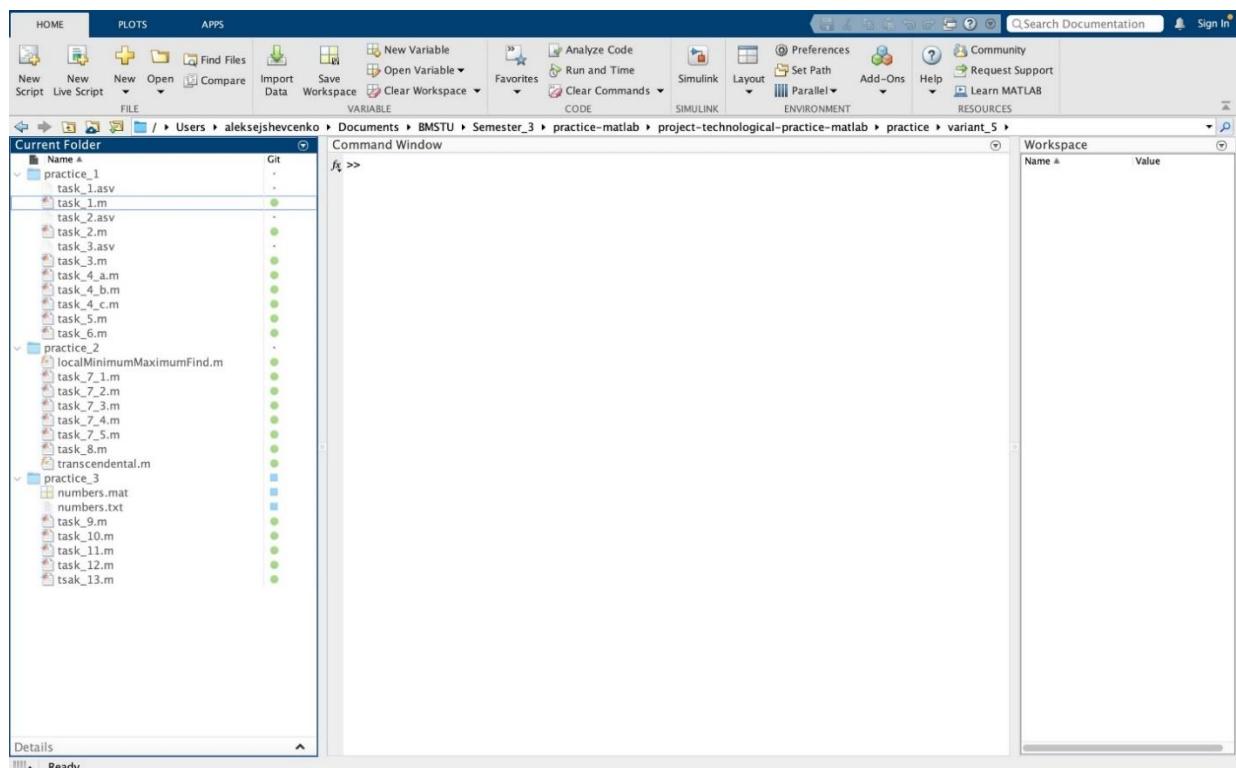


Рис. Общей структуры практики

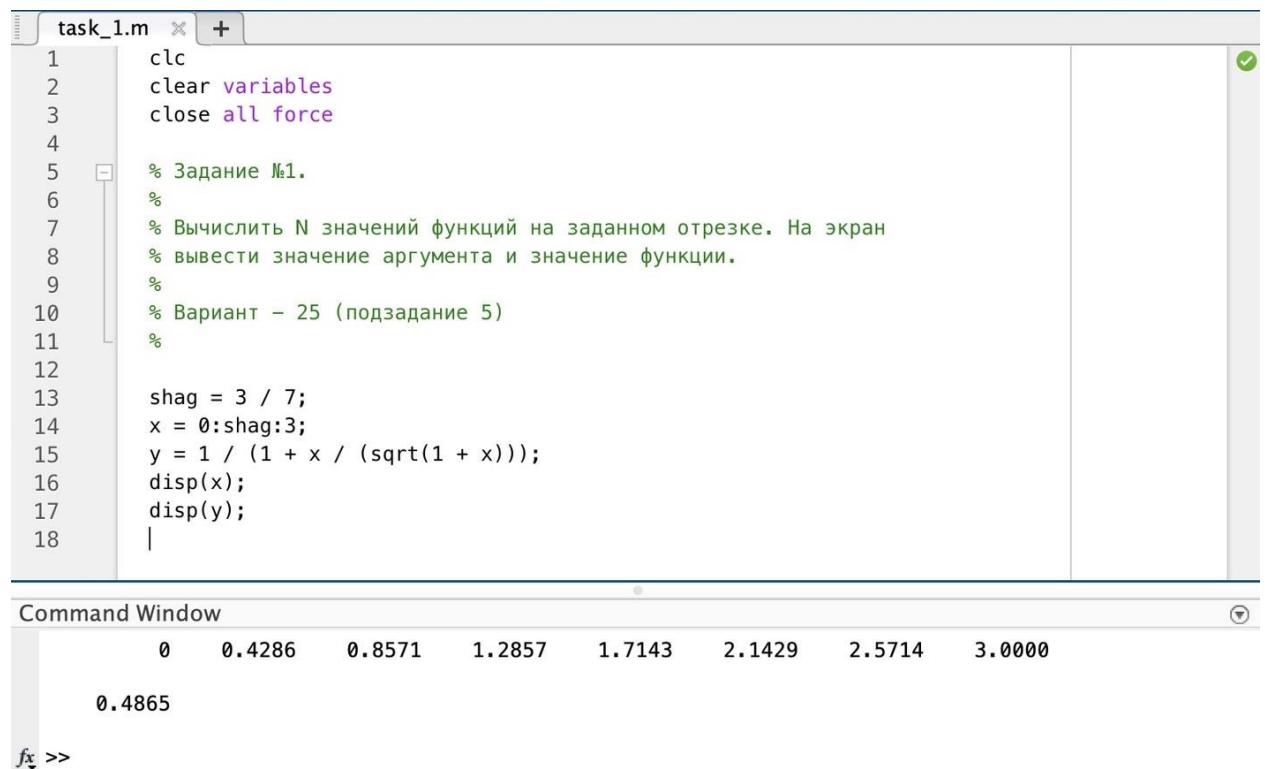
Практика 1

Задание 1

Задание №1. Вычислить N значений функции на заданном отрезке. На экран вывести значения аргумента и значения функции.

Варианты

$$5. \quad y(x) = \frac{1}{1 + \frac{x}{\sqrt{1+x}}} \quad [0,3] \quad N=8$$



The screenshot shows the MATLAB environment. At the top, there is a tab labeled "task_1.m". Below it, the code is displayed in a text editor:

```

1 clc
2 clear variables
3 close all force
4
5 % Задание №1.
6 %
7 % Вычислить N значений функций на заданном отрезке. На экран
8 % вывести значение аргумента и значение функции.
9 %
10 % Вариант – 25 (подзадание 5)
11 %
12
13 shag = 3 / 7;
14 x = 0:shag:3;
15 y = 1 / (1 + x / (sqrt(1 + x)));
16 disp(x);
17 disp(y);
18 |

```

Below the code editor is the "Command Window" pane. It shows the following output:

```

Command Window
0    0.4286    0.8571    1.2857    1.7143    2.1429    2.5714    3.0000
0.4865
fx >>

```

Рис. Код задачи 1 и вывод в результата

Задание 2

Задание №2. Для заданных векторов a и b длиной n (значения элементов векторов и их длину студент задает сам) выполнить преобразования и вычисления в соответствии с вариантом.

Варианты

5. Получить вектор x , содержащий удвоенные значения элементов вектора a , и вектор y , содержащий утроенные значения элементов вектора b . Определить среднее арифметическое каждого вектора.

```

task_1.m task_2.m +
1 clc
2 clear variables
3 close all force
4
5 % Задание №2
6 %
7 % Для заданных векторов a и b длиной n (значение элементов векторов и их
8 % длину студент задаёт сам) выполнить преобразования и вычисления в
9 % соответствии с вариантом.
10 %
11 % Вариант – 25 (подзадание 5)
12 %
13 % Получить вектор x, содержащий удвоенные значения элементов вектора a, и
14 % вектор y, содержащий утроенные значения элементов вектора b. Определить
15 % среднее арифметическое каждого вектора.
16 %
17
18 vector_1 = [1, 2, 3, 4, 5, 6, 7];
19 vector_2 = [1, 2, 3, 4, 5, 6, 7];
20
21 x = vector_1 * 2;
22 y = vector_2 * 3;
23
24 disp(x);
25 disp(y);
26
27 average_1 = sum(x) / length(x);
28 average_2 = sum(y) / length(y);
29
30 disp(average_1);
31 disp(average_2);
32 |

```

Command Window

```

3   6   9   12   15   18   21
8
12
>>

```

Рис. Код задачи 2 и вывод результата

Задание 3

Задание №3. При помощи встроенных функций для заполнения стандартных матриц, индексации двоеточием и, возможно, объединения, поворота или транспонирования получить следующие матрицы. Применить функции обработки данных и поэлементные операции для нахождения заданных величин.

$$5. \quad A = \begin{bmatrix} 1 & 1 & -3 & -3 & -3 & -3 \\ 1 & 1 & -3 & -3 & -3 & -3 \\ -3 & -3 & 2 & 0 & 0 & 0 \\ -3 & -3 & 0 & 2 & 0 & 0 \\ -3 & -3 & 0 & 0 & 2 & 0 \\ -3 & -3 & 0 & 0 & 0 & 2 \end{bmatrix} \quad x = \sum_{i=1}^6 a_{1i} + \sum_{j=1}^6 a_{4j}$$

```

task_1.m x task_2.m x task_3.m x +
1 clc
2 clear variables
3 close all force
4
5 % Задание №3
6 %
7 % При помощи встроенных функций для заполнения стандартных матриц,
8 % индексации двоеточием и, возможно, объединения, поворота или
9 % транспонирования получить следующие матрицы. Применить функции обработки
10 % и поэлементные операции для нахождения заданных величин.
11 %
12 % Вариант – 25 (подзадача 5)
13 %
14
15 A = zeros(6, 6);
16 element_unos = ones(2, 2);
17 element_minus = ones(1, 4);
18
19 element_minus = element_minus * (-3);
20
21 A(1:2, 1:2) = A(1:2, 1:2) + element_unos(1:2, 1:2);
22
23 A(1:2, 3:6) = A(1:2, 3:6) + element_minus(1, 1:4);
24
25 transp_element_minus = element_minus';
26
27 A(3:6, 1:2) = A(3:6, 1:2) + transp_element_minus(1:4, 1);
28
29 element_diagonale_6 = eye(6);
30 element_diagonale_2 = eye(2);
31
32 A = A + 2 * element_diagonale_6;
33 A(1:2, 1:2) = A(1:2, 1:2) - 2 * element_diagonale_2;
34
35 disp(A);
36
37 end_summa = sum(A(1, 1:6)) + sum(A(4, 1:6));
38
39 disp(end_summa);
40 |

```

Command Window

```

1   1   -3   -3   -3   -3
1   1   -3   -3   -3   -3
-3  -3   2    0    0    0
-3  -3   0    2    0    0
-3  -3   0    0    2    0
-3  -3   0    0    0    2
-14
fx >>

```

Рис. Код задачи 3 и вывод результата

Задание 4

Задание №4. Построить графики двух функций на заданном отрезке. Вывести графики:

- в разных окнах,
- в одном окне в одних осях,
- в одном окне в разных осях.

Использовать различные цвета, стили, подписи, легенду. Нанести сетку.

Варианты

$$5. \ f(x) = |2x|^3; \quad g(x) = |2x|^5; \quad x \in [-0.5, 0.5]$$

Пункт а)

```

task_1.m x task_2.m x task_3.m x task_4_b.m x task_4_a.m x +
1 clc
2 clear variables
3 close all force
4
5 % Задание №4
6 %
7 % Построить график двух функций на заданном отрезке. Вывести графики:
8 % - в разных окнах,
9 % - в одном окне в одних осях,
10 % - в одном окне в разных осях.
11 % Использовать различные цвета, стили, подписи, легенду. Нанести сетку.
12 %
13 % Вариант - 25 (подзадание 5)
14 %
15 % Построить график кусочно-заданной функции, отобразить ветви разными
16 % цветами и маркерами.
17 %
18 % a) В разных окнах
19 %
20 X = linspace(-0.5, 0.5, 100);
21 F = abs(2 * X).^3;
22 G = 2 * X.^5;
23 subplot(1, 2, 1)
24 plot(X, F, '+red-')
25 title('abs(2x)^3');
26 xlabel('ось x');
27 ylabel('ось y');
28 grid on
29 hold on
30 subplot(1, 2, 2);
31 plot(X, G, 'pblue:');
32 title('2x^5')
33 hold off
34 xlabel('ось x');
35 ylabel('ось y');
36 grid on
37
38

```

Рис. Код задачи 4

Построить график кусочно-заданной функции, отобразить ветви разными цветами и маркерами.

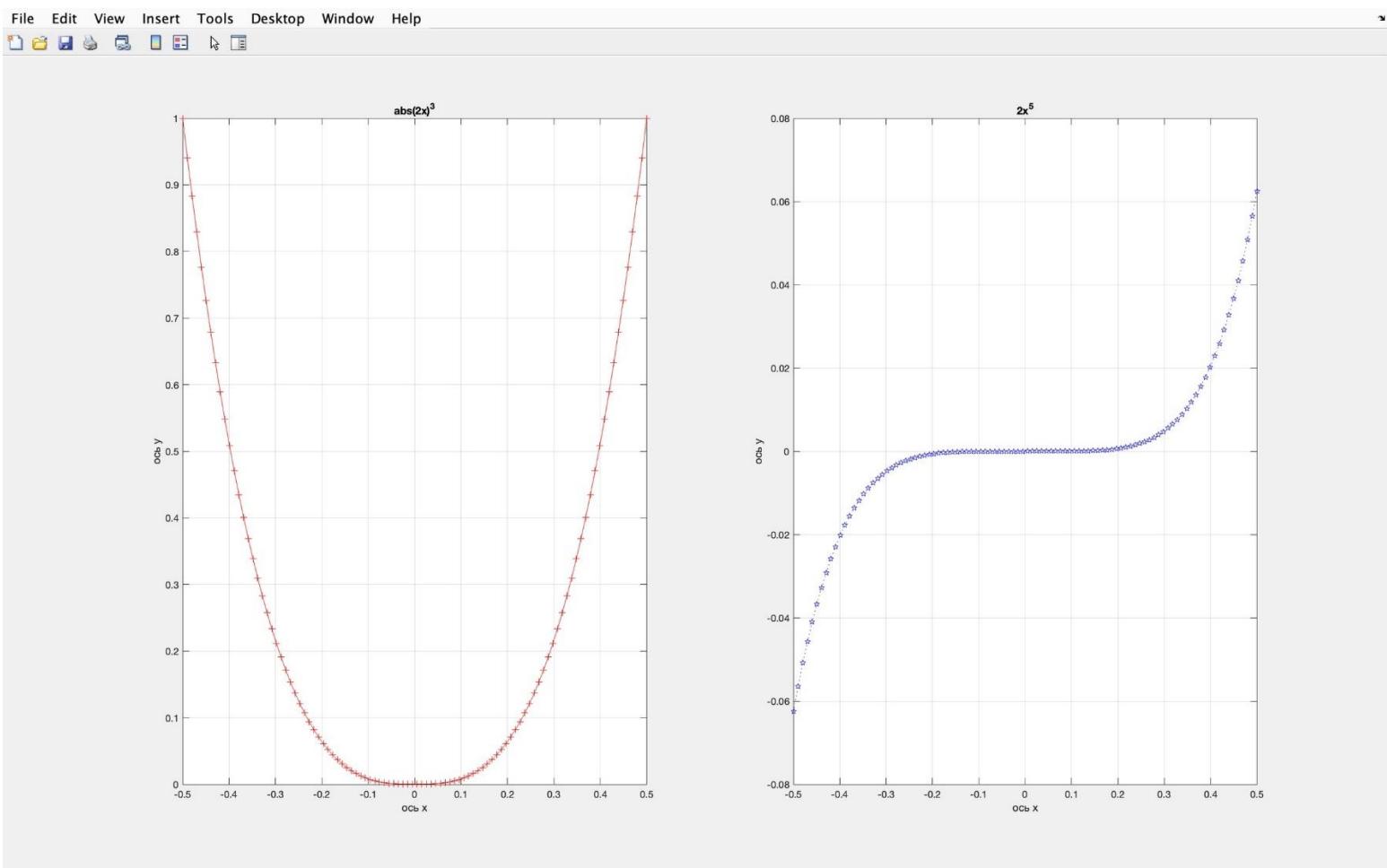
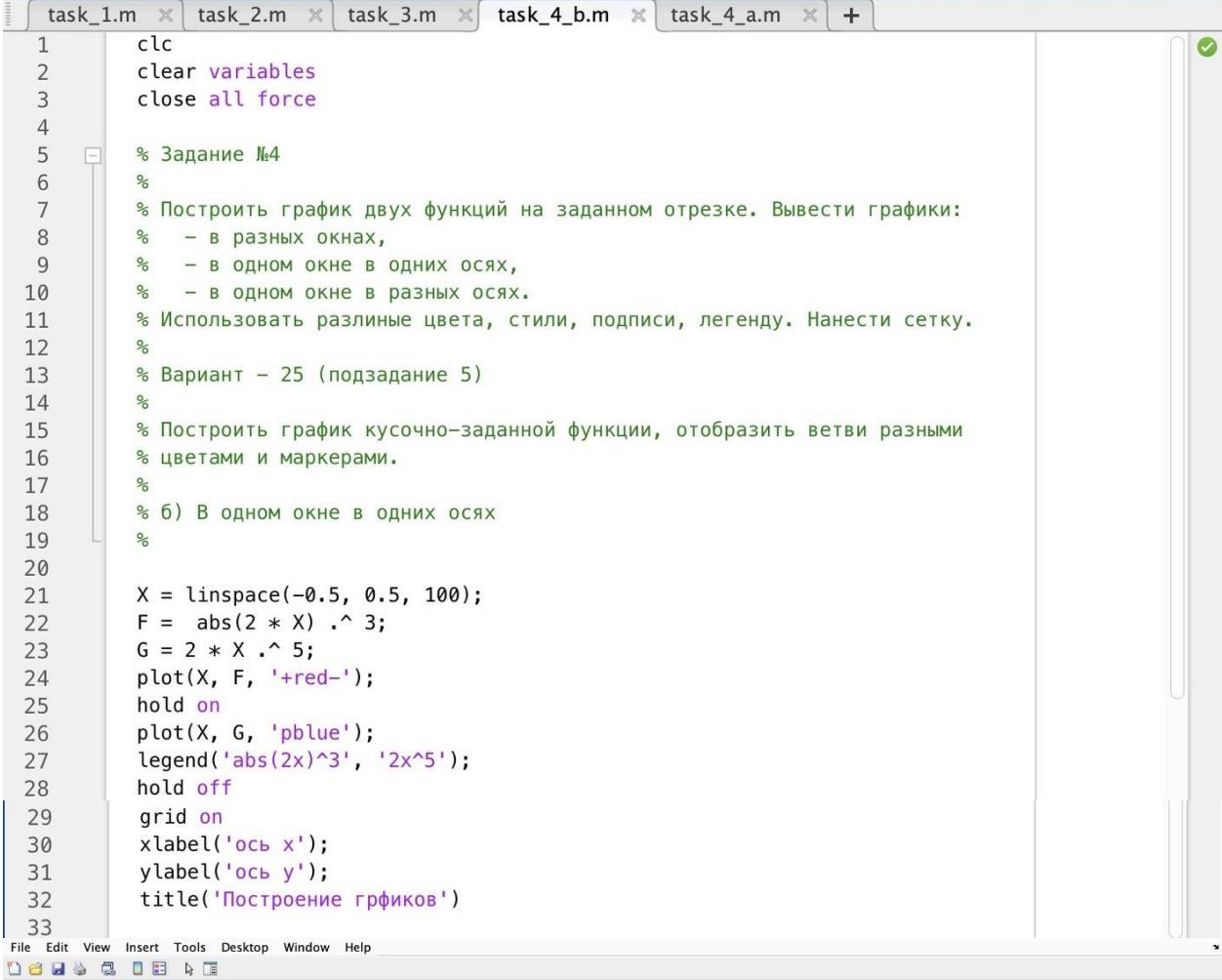


Рис. Раздельное отображение графиков задания 4 а)

Пункт б)



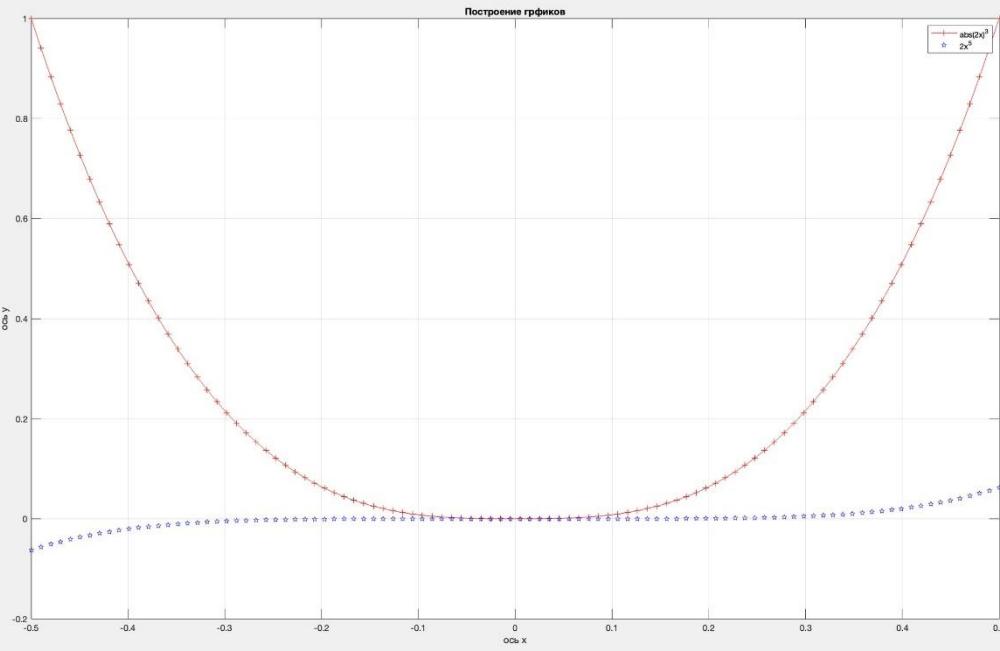
The screenshot shows a MATLAB interface with several tabs at the top: task_1.m, task_2.m, task_3.m, task_4_b.m (which is the active tab), task_4_a.m, and a plus sign tab. The code in the editor is as follows:

```

1 clc
2 clear variables
3 close all force
4
5 % Задание №4
6 %
7 % Построить график двух функций на заданном отрезке. Вывести графики:
8 % - в разных окнах,
9 % - в одном окне в одних осях,
10 % - в одном окне в разных осях.
11 % Использовать разные цвета, стили, подписи, легенду. Нанести сетку.
12 %
13 % Вариант – 25 (подзадание 5)
14 %
15 % Построить график кусочно-заданной функции, отобразить ветви разными
16 % цветами и маркерами.
17 %
18 % б) В одном окне в одних осях
19 %
20
21 X = linspace(-0.5, 0.5, 100);
22 F = abs(2 * X) .^ 3;
23 G = 2 * X .^ 5;
24 plot(X, F, '+red-');
25 hold on
26 plot(X, G, 'pblue');
27 legend('abs(2x)^3', '2x^5');
28 hold off
29 grid on
30 xlabel('ось x');
31 ylabel('ось y');
32 title('Построение графиков')
33

```

Below the editor is a menu bar: File, Edit, View, Insert, Tools, Desktop, Window, Help. Underneath the menu bar is a toolbar with icons for file operations.



The plot window is titled "Построение графиков". It displays two curves on a grid. The x-axis is labeled "ось x" and ranges from -0.5 to 0.5. The y-axis is labeled "ось y" and ranges from -0.0 to 1.0. The first curve, labeled "abs(2x)^3", is plotted with red '+' markers and a red line segment. It decreases from approximately 0.9 at x = -0.5 to 0 at x = 0. The second curve, labeled "2x^5", is plotted with blue 'x' markers and a blue line segment. It increases from approximately -0.05 at x = -0.5 to 0 at x = 0. A legend in the top right corner identifies the two series.

Рис. Код задания 4 б) и вывод графиков в одном окне в одних осях

Пункт в)

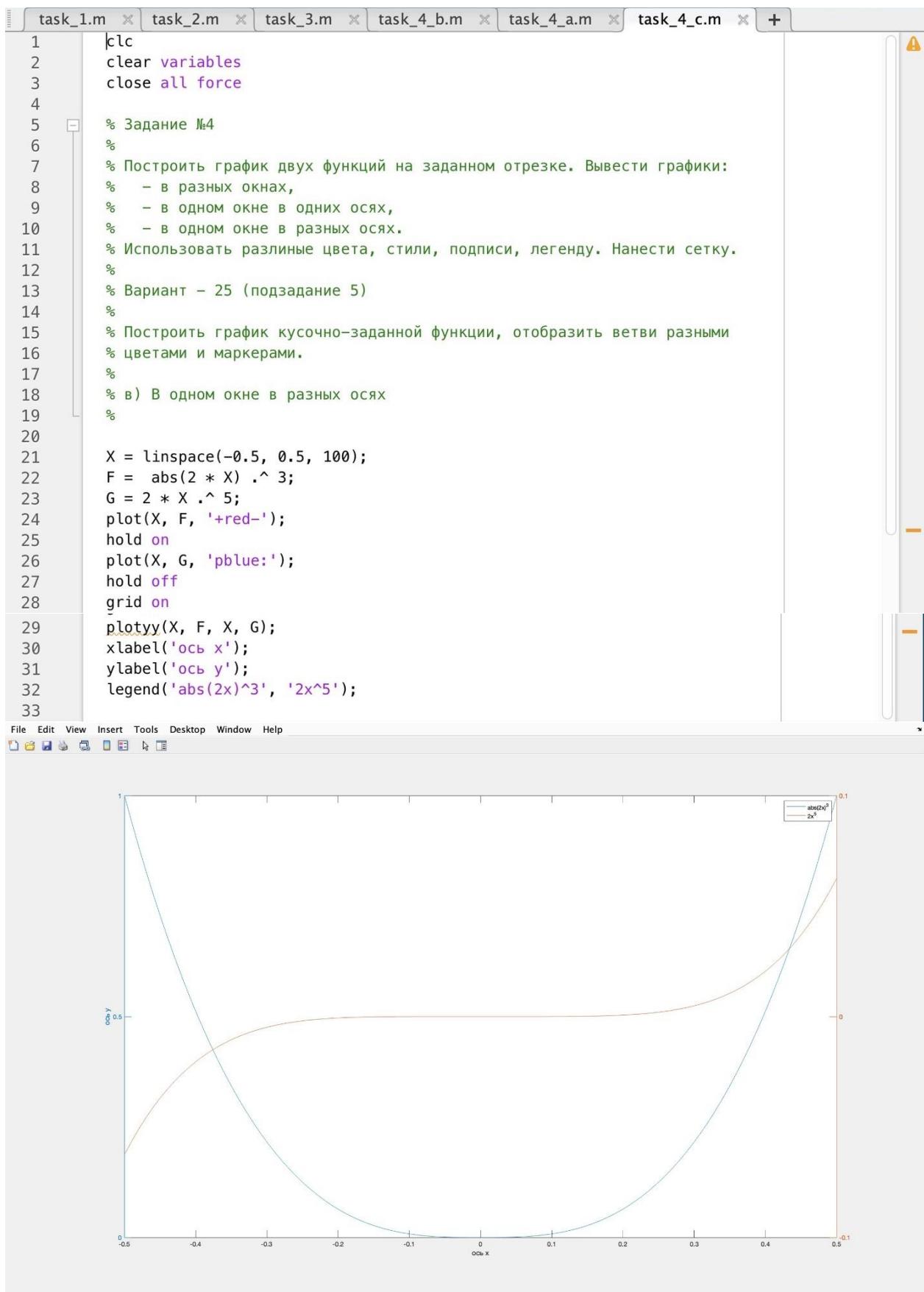


Рис. Код задания 4 в) и вывод графиков в одном окне в разных осях

```

x = -0.5: 5: 100;
f = abs(2.*x).^3;
g = abs(2.*x).^5;

figure(1)
scatter(x, f)
title('scatter')
legend('abs(2.*x).^3');
grid on

figure(2)
stem(x, g)
title('stem')
legend('abs(2.*x).^5');
grid on

figure(3)
stairs(x, f)
title('stairs')
legend('abs(2.*x).^3');
grid on

```

Рис. Код различных стилей отображения графиков задания 4

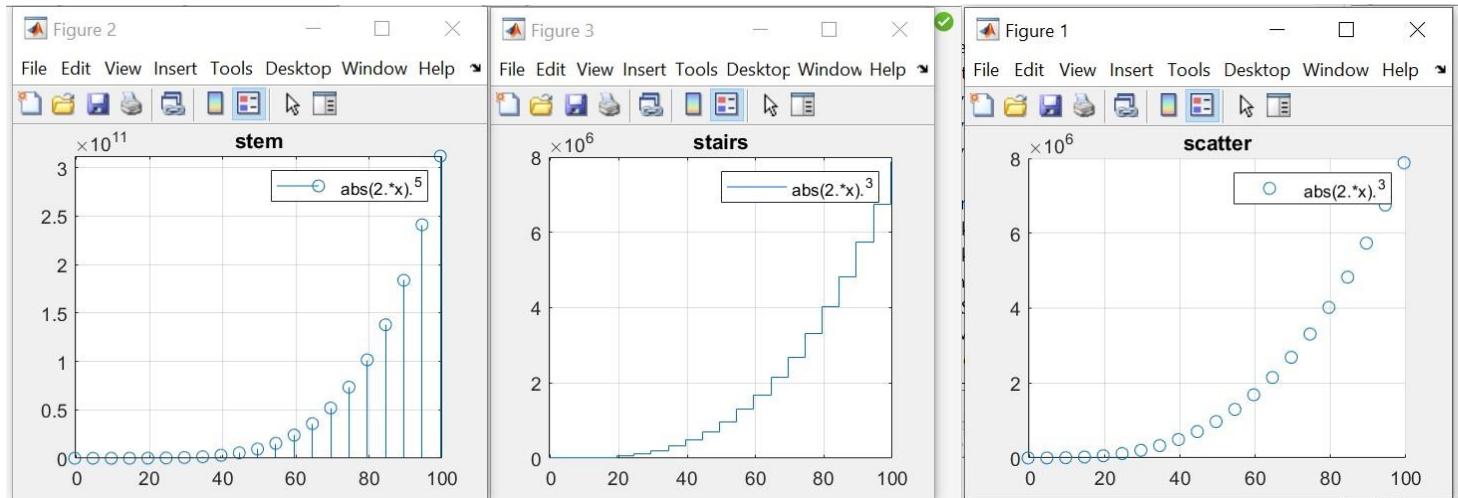


Рис. Примеры различных отображений графиков задания 4
(stem, stairs и scatter)

Задание 5

Построить график кусочно-заданной функции, отобразить ветви разными цветами и маркерами.

$$5. z(x, y) = \frac{\sin xy}{x} \quad x \in [0, 1, 5] \quad y \in [-\pi, \pi]$$

```

task_1.m task_2.m task_3.m task_4_b.m task_4_a.m task_4_c.m task_5.m +
1 clc
2 clear variables
3 close all force
4
5 % Задание №5
6 %
7 % Построить график функций двух переменных.
8 %
9 % Вариант – 25 (подзадание 5)
10 %
11
12 figure(14)
13 [X, Y] = meshgrid(0.1:0.245:5, -pi:pi/10:pi);
14 Z = sin(X.* Y) ./ X;
15 subplot(2, 2, 1);
16 surf(X, Y, Z);
17 subplot(2, 2, 2);
18 surfc(X, Y, Z);
19 subplot(2, 2, 3);
20 surfl(X, Y, Z);
21 subplot(2, 2, 4);
22 surface(X, Y, Z)
23 subplot(2, 2, 4)
24 s.EdgeColor = 'none'
25

```

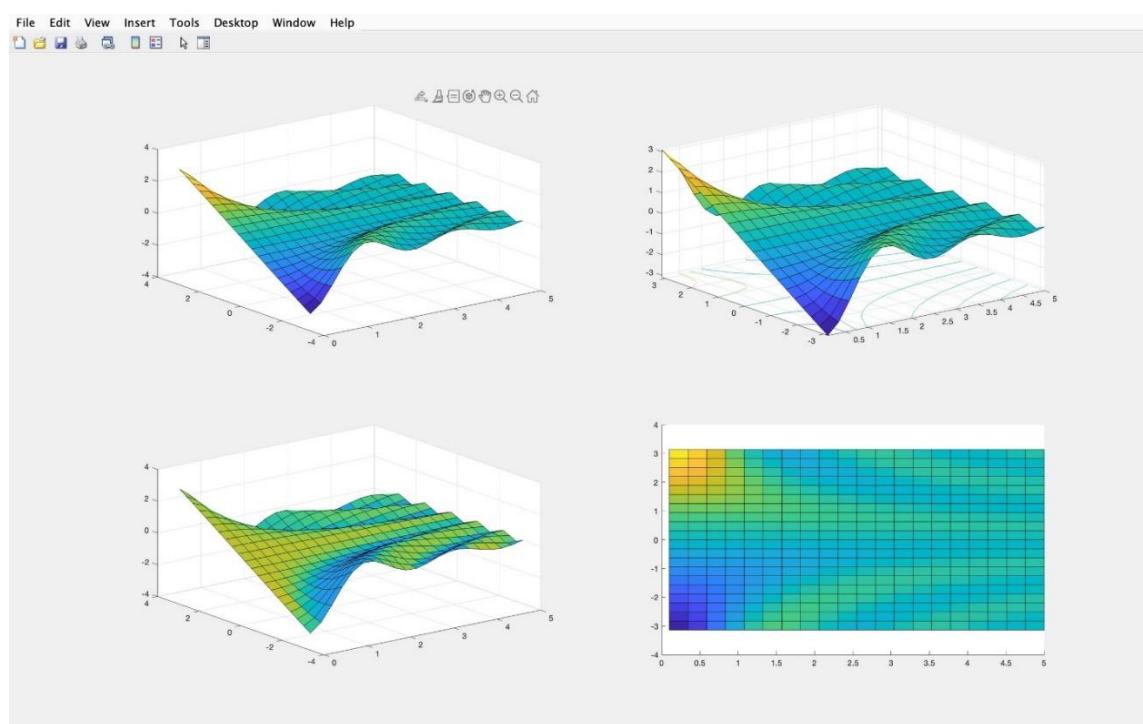


Рис. Код задания 5 и вывод графиков в 3-х мерном пространстве

```

2      X=0.1:0.245:5;
3      Y=-pi:pi/10:pi;
4      Z=sin(X.*Y)./X;
5      figure(1)
6      plot3(X, Y, Z)
7
8      grid on
9      axis equal
10
11     xlabel('X')
12     ylabel('Y')
13     zlabel('Z')
14
15     figure(2)
16     scatter3(X, Y, Z)
17
18     grid on
19     axis equal
20
21     xlabel('X')
22     ylabel('Y')
23     zlabel('Z')

```

Рис. Код отображения графика функции двух переменных задания 5

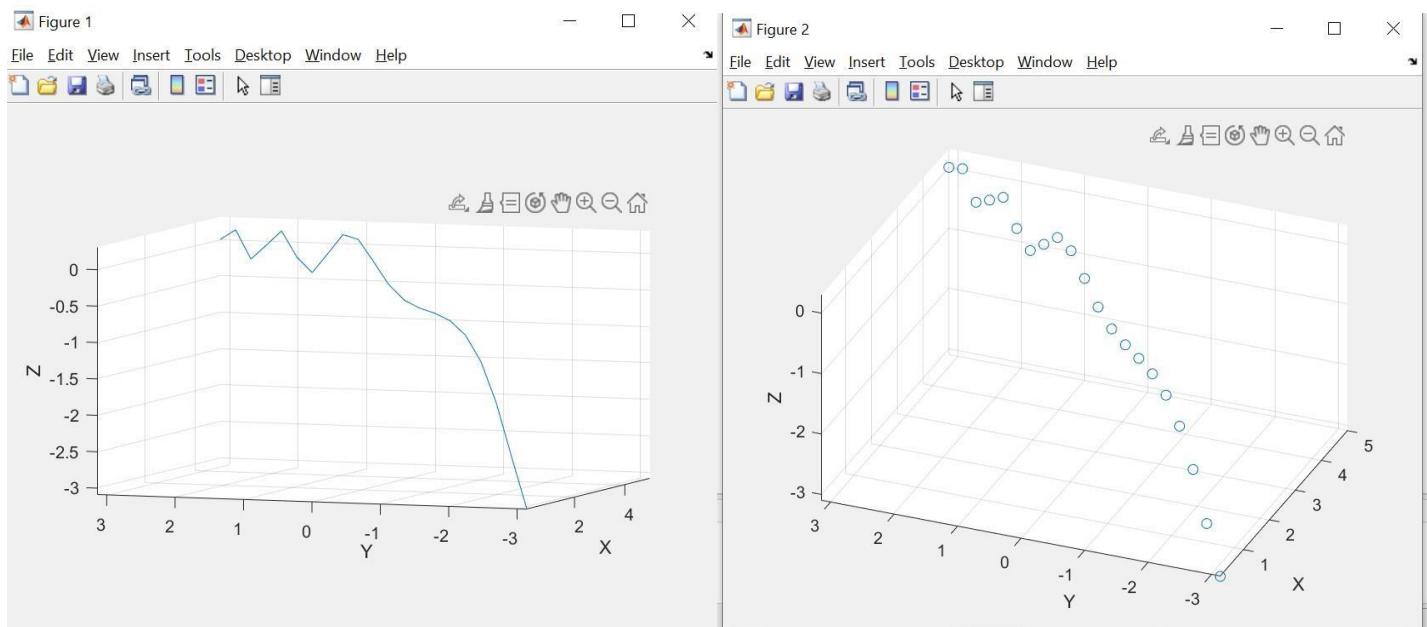


Рис. Графики отображения функции двух переменных задания 5

Задание 6

Задание №6. Написать скрипт-файл для решения следующих задач.

Варианты

5. Сложить все элементы заданной матрицы, кроме элементов главной диагонали.

The screenshot shows the MATLAB environment. The code editor window at the top contains the following script:

```
+1 task_2.m task_3.m task_4_b.m task_4_a.m task_4_c.m task_5.m task_6.m
1 clc
2 clear variables
3 close all force
4
5 % Задание №6
6 %
7 % Написать скрипт-файл для решения следующих задач.
8 % Сложить все элементы заданной матрицы, кроме элементов главной диагонали.
9 %
10 % Вариант - 25 (подзадание 5)
11 %
12
13 A = [1 2 3; 4 5 6; 7 8 9]
14 S = sum(A(:)) - sum(diag(A))
15
```

The command window below displays the execution results:

```
Command Window
A =
1 2 3
4 5 6
7 8 9

S =
30
fx >>
```

Рис. Код и вывод задания 6

Практика 2

Задание 7

При написании программ обязательно:

- использовать комментарии, содержащие назначение программы и описание ее переменных;
- вывод результатов сопровождать пояснительным текстом.

По результатам работы должен быть составлен отчет, содержащий текст индивидуального задания, тексты *script*-файлов и файлов-функций, а также графическое представление результатов работы.

Варианты индивидуальных заданий

Вариант 5

1. Вычислить корни полинома

$$x^3 - 0,1x^2 + 1,5x - 1,5$$

2. Найти решение системы линейных уравнений

$$\begin{cases} 2,7x_1 + 0,9x_2 - 1,5x_3 = 3,5 \\ 4,5x_1 - 2,8x_2 + 6,7x_3 = 2,6 \\ 5,1x_1 + 3,7x_2 - 1,4x_3 = -0,14 \end{cases}$$

3. Найти значение локального минимума и максимума функции

$$f(x) = 0,5 \sin x - 0,2 \cos x$$

$$0 \leq x \leq \pi/3$$

4. Вычислить значение определенного интеграла

$$\int_{0,8}^{1,4} \frac{dx}{\sqrt{2x^2 + 3}}$$

5. Решить трансцендентное уравнение

$$e^x + \sqrt{1+e^{2x}} - 2 = 0$$

Задание 7

Пункт 1)

The screenshot shows the MATLAB environment. The code editor window is titled "task_7_1.m" and contains the following MATLAB script:

```
1 clc
2 clear variables
3 close all force
4
5 % Задание №7 (пункт № 1)
6 %
7 % При написании программ обязательно:
8 %   - использовать комментарии, содержащие назначение программы и описание
9 %     её переменных;
10 %   - вывод результатов сопровождать пояснительным текстом.
11 % По результатам работы должен быть составлен отчёт, содержащий текст
12 % индивидуального задания, тексты script-файлов и файлов-функций, а так же
13 % графическое представление результатов.
14 %
15 % Вариант – 25 (подзадание 5)
16 %
17 % 1. Вычислить корни полинома
18 %       $x^3 - 0,1x^2 + 1,5x - 1,5$ 
19 %
20
21 p = [1 -0.1 1.5 1.5];
22 r = roots(p);
23 disp(r);
```

The command window below shows the output of the script:

```
0.4092 + 1.3858i
0.4092 - 1.3858i
-0.7184 + 0.0000i
```

Рис. Код и вывод задания 7 пункт 1)

Задание 7

Пункт 2)

The screenshot shows the MATLAB environment. In the top window, titled 'task_7_1.m' (which is currently active), there is a script containing MATLAB code. The code includes comments explaining the requirements for the task, such as using comments for program description and outputting results with text. It also defines matrices A and B and uses the linsolve function to solve the system of linear equations.

```

1 clc
2 clear variables
3 close all force
4
5 % Задание №7 (пункт № 2)
6 %
7 % При написании программ обязательно:
8 %   - использовать комментарии, содержащие назначение программы и описание
9 %     её переменных;
10 %   - вывод результатов сопровождать пояснительным текстом.
11 % По результатам работы должен быть составлен отчёт, содержащий текст
12 % индивидуального задания, тексты script-файлов и файлов-функций, а так же
13 % графическое представление результатов.
14 %
15 % Вариант – 25 (подзадание 5)
16 %
17 % 2. Найти решение системы линейных уравнений
18 %      (1) 2,7 * x1 + 0,9 * x2 - 1,5 * x3 = 3,5
19 %      (2) 4,5 * x1 - 2,8 * x2 + 6,7 * x3 = 2,6
20 %      (3) 5,1 * x1 + 3,7 * x2 - 1,4 * x3 = -0.14
21 %
22
23 A = [2.7 0.9 1.5; 4.5 -2.8 6.7; 5.1 3.7 -1.4];
24 B = [3.5 ; 2.6 ; -0.14];
25
26 x = linsolve(A, B);
27 disp(x);
28 |

```

In the bottom window, titled 'Command Window', the output of the code execution is shown:

```

-2.1448
4.2886
3.6209

```

Рис. Код и вывод задания 7 пункт 2)

Задание 7

Пункт 3)

The screenshot shows the MATLAB interface. In the code editor, the file `task_7_3.m` is open, containing the following code:

```

1 clc
2 clear variables
3 close all force
4
5 % Задание №7 (пункт № 3)
6 %
7 % При написании программ обязательно:
8 %   - использовать комментарии, содержащие назначение программы и описание
9 %     её переменных;
10 %   - вывод результатов сопровождать пояснительным текстом.
11 % По результатам работы должен быть составлен отчёт, содержащий текст
12 % индивидуального задания, тексты script-файлов и файлов-функций, а так же
13 % графическое представление результатов.
14 %
15 % Вариант – 25 (подзадание 5)
16 %
17 % 3. Найти значение локального минимума и максимума функции
18 %      f(x) = 0,5 * sin(x) – 0,2 * cos(x)
19 %      0 <= cos(x) <= pi / 3
20 %
21 [x, y] = fminbnd(@localMinimumMaximumFind, 1, 3)
22
23

```

In the Command Window, the output is:

```

Command Window
x =
1.1637
y =
-0.7896
fx >>

```

Рис. Код и вывод задания 7 пункт 3)

Использовали функцию `localMinimumMaximumFind`

The screenshot shows the MATLAB interface with the code editor open to the file `localMinimumMaximumFind.m`. The code is as follows:

```

+3 task_7_4.m x task_7_5.m x task_8.m x transcendental.m x localMinimumMaximumFind.m x +
1 function y = localMinimumMaximumFind(x)
2
3 %
4 % Функция нахождения локального минимума функции вида:
5 %      f(x) = - (2 * sin(x .^ 2) – x);
6 %      на отрезке 0 <= x <= pi/3
7 %
8
9 y = - (2 * sin(x .^ 2) – x);
10
11 end
12

```

Рис. Код вспомогательной функции `localMinimumMaximumFind`

Задание 7

Пункт 4)

The screenshot shows the MATLAB interface. In the top navigation bar, there are tabs for 'task_7_1.m', 'task_7_2.m', 'task_7_3.m', and 'task_7_4.m'. The 'task_7_4.m' tab is active. Below the tabs is a code editor window containing the following MATLAB script:

```
1 clc
2 clear variables
3 close all force
4
5 % Задание №7 (пункт № 4)
6 %
7 % При написании программ обязательно:
8 %   - использовать комментарии, содержащие назначение программы и описание
9 %     её переменных;
10 %   - вывод результатов сопровождать пояснительным текстом.
11 % По результатам работы должен быть составлен отчёт, содержащий текст
12 % индивидуального задания, тексты script-файлов и файлов-функций, а так же
13 % графическое представление результатов.
14 %
15 % Вариант – 25 (подзадание 5)
16 %
17 % 4. Вычислить значение определённого интеграла (методом Симпсона)
18 %
19
20 expression = '1 ./ ((2 .* x .^ 2 + 3).^ (1/2))';
21 disp(quad(expression, 0.8, 1.4, 0.0001));
```

Below the code editor is the 'Command Window' pane, which displays the output:

```
0.2582
```

Рис. Код и вывод задания 7 пункт 4)

Задание 7

Пункт 5)

The screenshot shows the MATLAB interface. In the code editor, several files are listed at the top: task_7_1.m, task_7_2.m, task_7_3.m, task_7_4.m, task_7_5.m, and a new file icon. The task_7_5.m file is open, containing the following code:

```

1 clear variables
2 close all force
3
4
5 % Задание №7 (пункт № 5)
6 %
7 % При написании программ обязательно:
8 %   - использовать комментарии, содержащие назначение программы и описание
9 %     её переменных;
10 %   - вывод результатов сопровождать пояснительным текстом.
11 % По результатам работы должен быть составлен отчёт, содержащий текст
12 % индивидуального задания, тексты script-файлов и файлов-функций, а так же
13 % графическое представление результатов.
14 %
15 % Вариант – 25 (подзадание 5)
16 %
17 % 5. Решить трансцендентное уравнение
18 %
19
20 input_function = @transcendental;
21 x0 = 1;
22 result_function = fzero(input_function, x0);
23 disp(result_function);
24

```

In the Command Window below, the output is shown:

```

Command Window
-0.2877
fx >>

```

Рис. Код задания и вывод результата задания 7 пункт 5)

Использовали функцию transcendental

The screenshot shows the MATLAB interface with the code editor open to a file named transcendental.m. The code is as follows:

```

+1 task_7_2.m task_7_3.m task_7_4.m task_7_5.m task_8.m transcendental.m *
1 function y = transcendental(x)
2 %
3 %
4 % Функция нахождения локального минимума функции вида:
5 %   f(x) = - (2 * sin(x .^ 2) - x);
6 %   на отрезке 0 <= x <= pi/3
7 %
8
9 y = exp(x) + (1 + exp(2 * x)) ^ (1 / 2) - 2;
10
11 end
12

```

Рис. Код вспомогательной функции transcendental

Задание 8

Вариант 5. Построить траекторию стартующей ракеты в трехмерном пространстве. Законы изменения координат:

$$\left. \begin{array}{l} x(t) = x_0 + a_{x1} \frac{t^2}{2}, \\ y(t) = y_0, \\ h(t) = a_{h1} \frac{t^2}{2}, \end{array} \right\} \text{при } 0 \leq t \leq 10;$$

$$\left. \begin{array}{l} x(t) = x(10) + a_{x2} \frac{t^2}{2}, \\ y(t) = y_0 + a_y \frac{t^2}{2}, \\ h(t) = a_{h2} \frac{t^2}{2} + h(10) \end{array} \right\} \text{при } 10 < t \leq 50,$$

где $x_0 = 5$ км; $y_0 = 5$ км; $a_{x1} = 20$ м/с²; $a_{x2} = 10$ м/с²; $a_{h1} = 40$ м/с²; $a_y = 20$ м/с²; $a_{h2} = 0,3a_y$.

Воспользоваться оператором *plot3*, местоположение ракеты указывать красной звездочкой. С помощью операторов *line* показать проекции точек положения летательного аппарата на горизонтальную и вертикальную плоскости. Шаг изменения времени $\Delta t = 1$ с.

```

1 clc
2 clear variables
3 close all force
4
5 % Задание №8
6 %
7 % При написании программ обязательно:
8 % - использовать комментарии, содержащие назначение программы и описание
9 % её переменных;
10 % - вывод результатов сопровождать пояснительным текстом.
11 % По результатам работы должен быть составлен отчёт, содержащий текст
12 % индивидуального задания, тексты script-файлов и файлов-функций, а так же
13 % графическое представление результатов.
14 %
15 % Вариант – 25 (подзадание 5)
16 %
17 % Построить траекторию стартующей ракеты в трёхмерном пространстве. Законы
18 % измерения координаты даны.
19 % Воспользоваться оператором plot3, местоположение ракеты указывать красной
20 % звёздочкой. С помощью операторов line показать проекции точек
21 % летательного аппарата на горизонтальную и вертикальную плоскости. Шаг
22 % измерения времени dt = 1с.
23 %
24 |
25 figure(1)
26
27 t = 0 : 1 : 10;
28 xt = 5 + 20 .* t .^ 2 / 2;
29 yt = 5 .* t .^ 0;
30 ht = 40 .* t .^ 2 ./ 2;
31
32 k = 11: 1 : 50;
33 xk = 1005 + 10 .* k .^ 2 ./ 2;
34 yk = 5 + 20 .* k .^ 2 ./ 2;
35 hk = 6 .* k .^ 2 ./ 2 + 2000;
36
37 plot3 (xt, yt, ht, xk, yk, hk, '.r')
38 grid on
39
40 xline(13505, '--');
41 yline(25005, '--');
42 zline(9500, '--');
43 |

```

Рис. Код задания 8

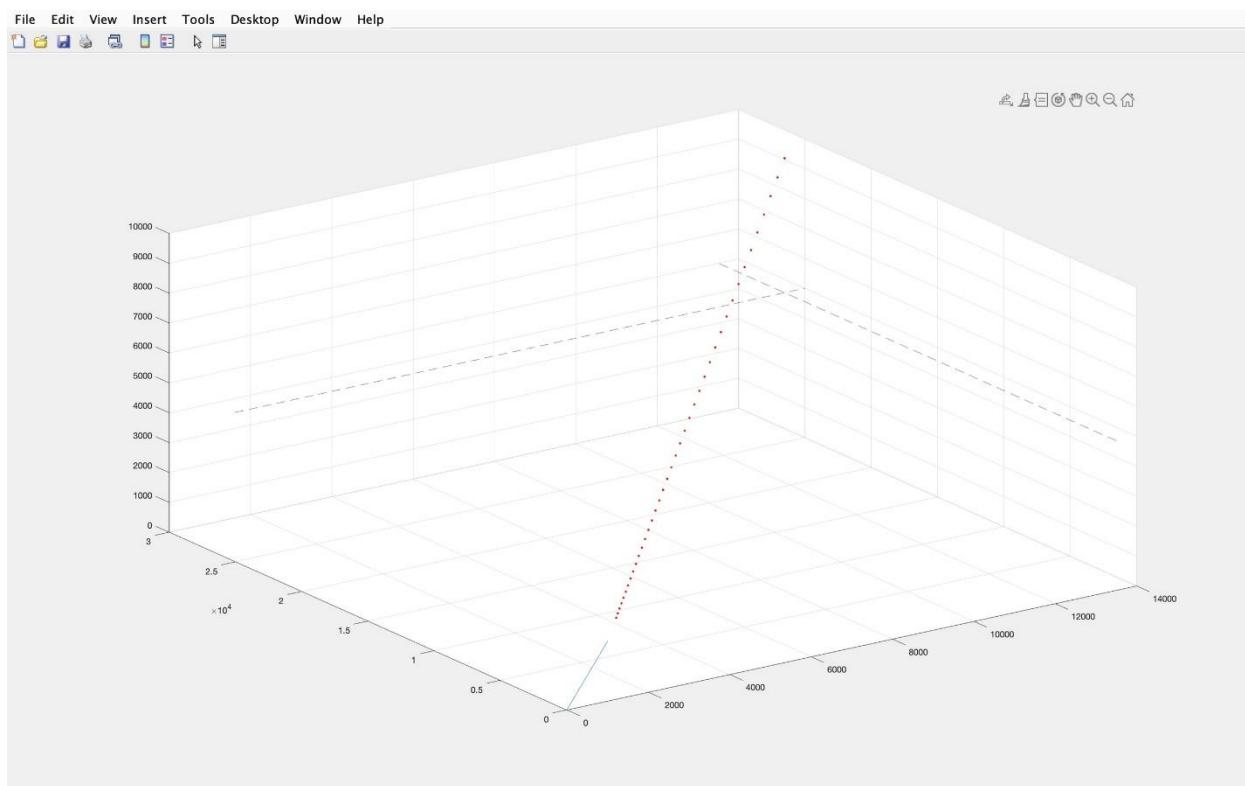


Рис. График траектории стартующей ракеты в 3-х мерном пространстве

Задания 8

Практика 3

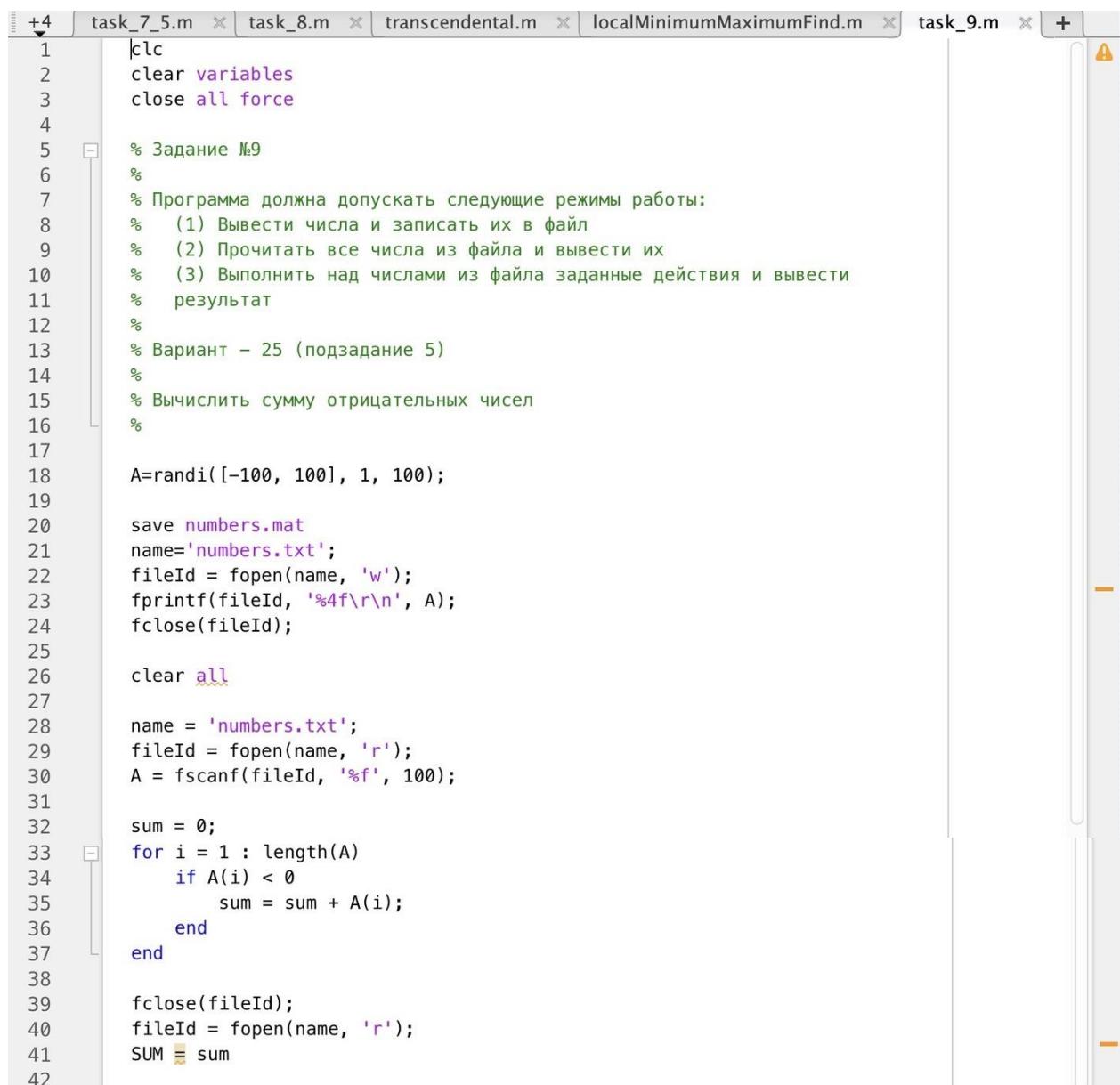
Задание 9

14. Файл последовательного доступа.

Программа должна допускать следующие режимы работы:

- (1) Ввести числа и записать их в файл
- (2) Прочитать все числа из файла и вывести их
- (3) Выполнить над числами из файла заданные действия и вывести результат

14.5. Вычислить сумму отрицательных чисел.



```

1 clc
2 clear variables
3 close all force
4
5 % Задание №9
6 %
7 % Программа должна допускать следующие режимы работы:
8 % (1) Вывести числа и записать их в файл
9 % (2) Прочитать все числа из файла и вывести их
10 % (3) Выполнить над числами из файла заданные действия и вывести
11 % результат
12 %
13 % Вариант – 25 (подзадание 5)
14 %
15 % Вычислить сумму отрицательных чисел
16 %
17
18 A=randi([-100, 100], 1, 100);
19
20 save numbers.mat
21 name='numbers.txt';
22 fileId = fopen(name, 'w');
23 fprintf(fileId, '%4f\r\n', A);
24 fclose(fileId);
25
26 clear all
27
28 name = 'numbers.txt';
29 fileId = fopen(name, 'r');
30 A = fscanf(fileId, '%f', 100);
31
32 sum = 0;
33 for i = 1 : length(A)
34     if A(i) < 0
35         sum = sum + A(i);
36     end
37 end
38
39 fclose(fileId);
40 fileId = fopen(name, 'r');
41 SUM = sum
42

```

Рис. Код задания 9

Задание 10

Найти производные функций:

849. $y = \frac{(1-x)^p}{(1+x)^q}.$

The screenshot shows the MATLAB environment. At the top, there is a tab bar with several files: task_9.m, task_10.m (which is currently selected), task_11.m, task_12.m, tsak_13.m, and numbers.txt. Below the tabs is a code editor window containing the following MATLAB script:

```

1 clc
2 clear variables
3 close all force
4
5 %
6 % Задание №10
7 %
8 % Найти производные функций: № 849
9 %
10 % Вариант – 25 (подзадание 5)
11 %
12
13 syms x y p q
14 y = (1 - x) ^ p / (1 + x) ^ q;
15 disp('dy / dx = '), disp(diff(y, x))
16

```

Below the code editor is the Command Window, which displays the results of running the script:

```

Command Window
dy / dx =
- (p*(1 - x)^(p - 1))/(x + 1)^q - (q*(1 - x)^p)/(x + 1)^(q + 1)
fx >>

```

Рис. Код и вывод результата задания 10

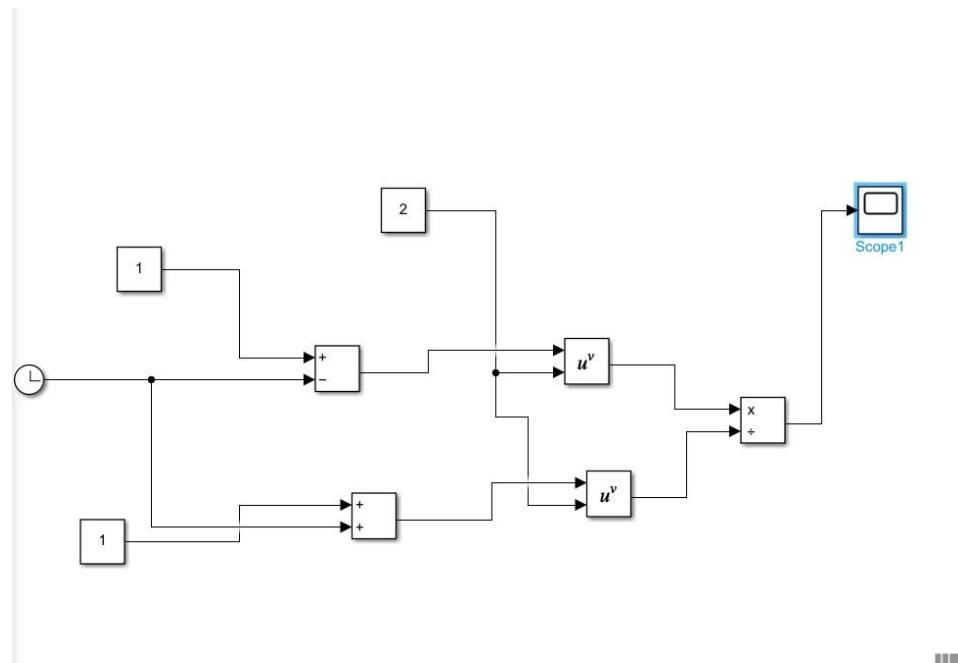


Рис. Схема Simulink решения отображения графика производной задания 10

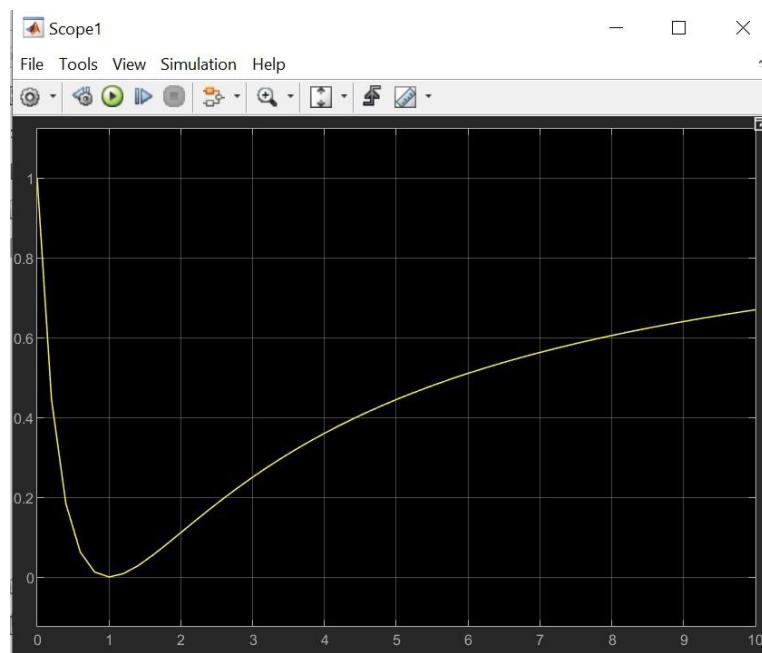


Рис. Отображение графика решения производной задания 10

Задание 11

найти следующие интегралы:

1870. $\int \frac{x^4}{x^4 + 5x^3 + 4} dx.$

The screenshot shows the MATLAB environment. At the top, there are tabs for several files: task_9.m, task_10.m, task_11.m (which is the active file), task_12.m, tsak_13.m, and numbers.txt. Below the tabs is a code editor window containing the following MATLAB script:

```

1 clc
2 clear variables
3 close all force
4
5 %
6 % Задание №11
7 %
8 % Найти следующие интегралы: № 1870
9 %
10 % Вариант – 25 (подзадание 5)
11 %
12
13 syms x y
14 y = x ^ 4 / (x ^ 4 + 5 * x ^ 2 + 4);
15 disp('result = ')
16 disp(int(y, x))

```

Below the code editor is the Command Window, which displays the result of running the script:

```

result =
x - (8*atan(x/2))/3 + atan(x)/3

```

Рис. Код и вывод результата задания 11

Задание 12

Вычислить интегралы:

2272.
$$\int_{-2}^{-1} \frac{dx}{x \sqrt{x^2 - 1}}.$$

The screenshot shows the MATLAB environment. At the top, there is a tab bar with several files: task_9.m, task_10.m, task_11.m, task_12.m (which is currently selected), tsak_13.m, and numbers.txt. Below the tabs is a code editor window containing the following MATLAB script:

```

1 clc
2 clear variables
3 close all force
4
5 %
6 % Задание №12
7 %
8 % Вычислить интеграл: № 2272
9 %
10 % Вариант – 25 (подзадание 5)
11 %
12
13 syms x y
14 y = 1 / (x * sqrt(x ^ 2 - 1));
15 disp('result = ')
16 disp(int(y, x, -2, -1))

```

Below the code editor is the Command Window, which displays the result of the execution:

```

Command Window
result =
-pi/3
f1 >> |

```

Рис. Код и вывод результата задания 12

Задание 13

Построить графики дробных рациональных функций:

257. $y = \frac{2x}{1+x^2}$ (серпантин Ньютона).

```

+8 task_9.m x task_10.m x task_11.m x task_12.m x tsak_13.m x +
1 clc
2 clear variables
3 close all force
4
5 %
6 % Задание №13
7 %
8 % Построить графики подобных рациональных функций: № 257
9 % Серпантин Ньютона
10 %
11 % Вариант – 25 (подзадание 5)
12 %
13
14 syms x y
15 y = 2 * x / (1 + x^2);
16 ezplot(y), grid on, ylabel('y')
17

```

Рис. Код задания 13

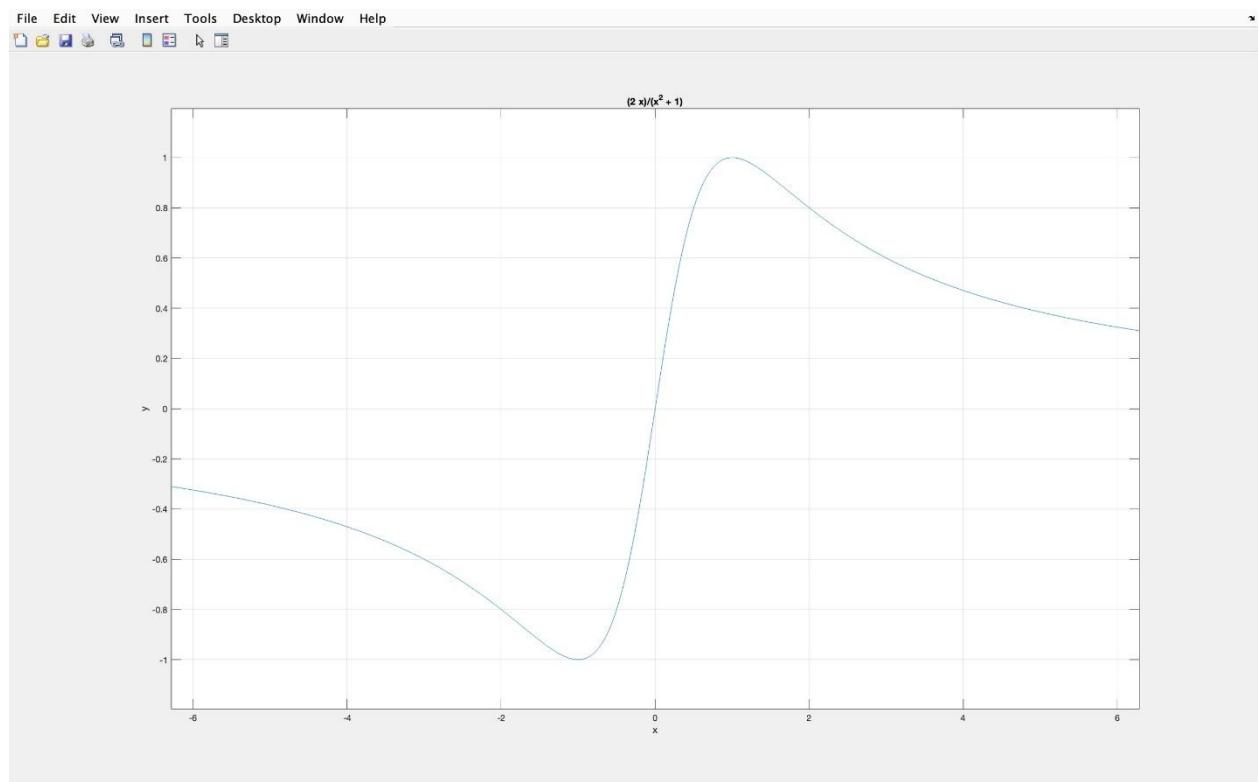


Рис. График дробных рациональных функций задания 13

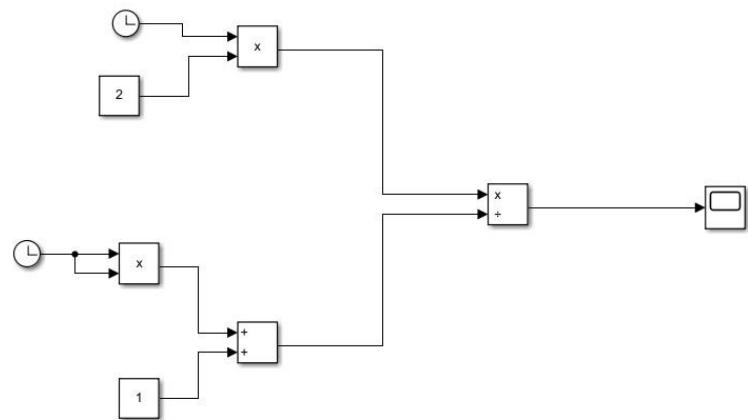
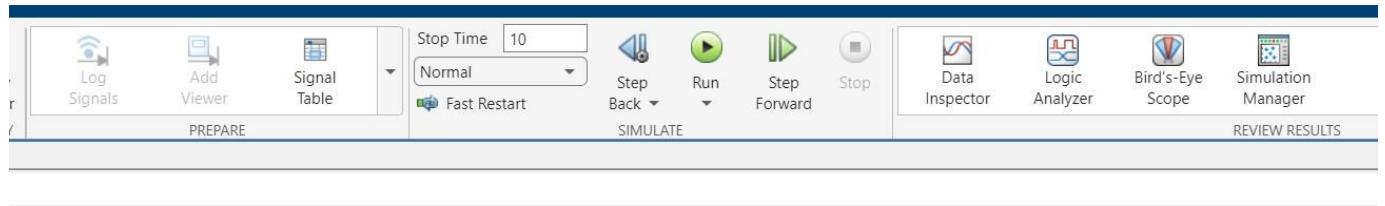


Рис. Схема Simulink решения дробных рациональных функций
Задания 13

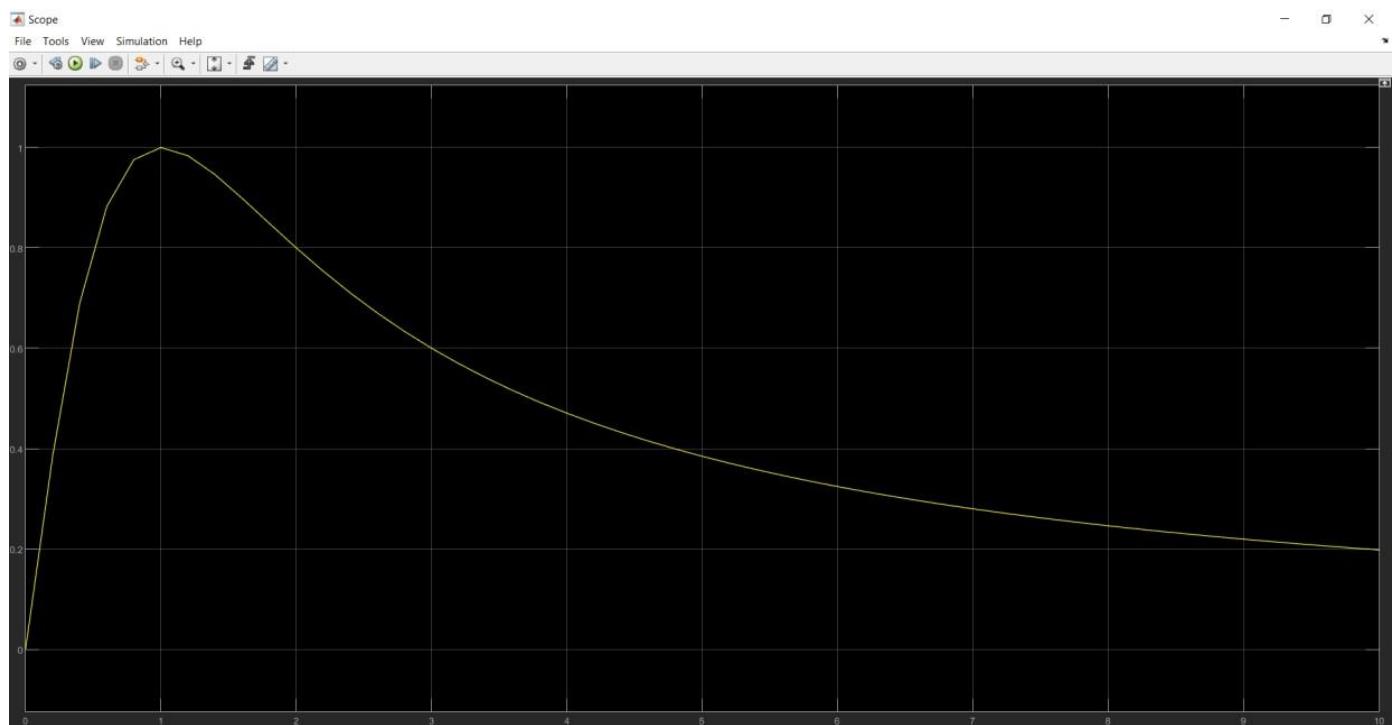


Рис. График отображения решения дробных рациональных функций задания 13

Задание 14

Построить модель следующей системы в Simulink

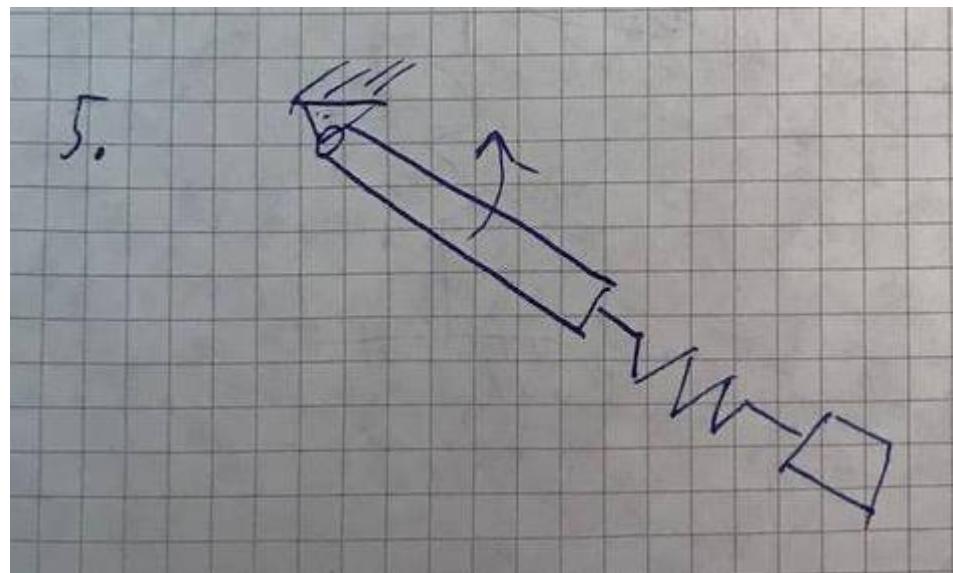


Рис. Иллюстрация модели

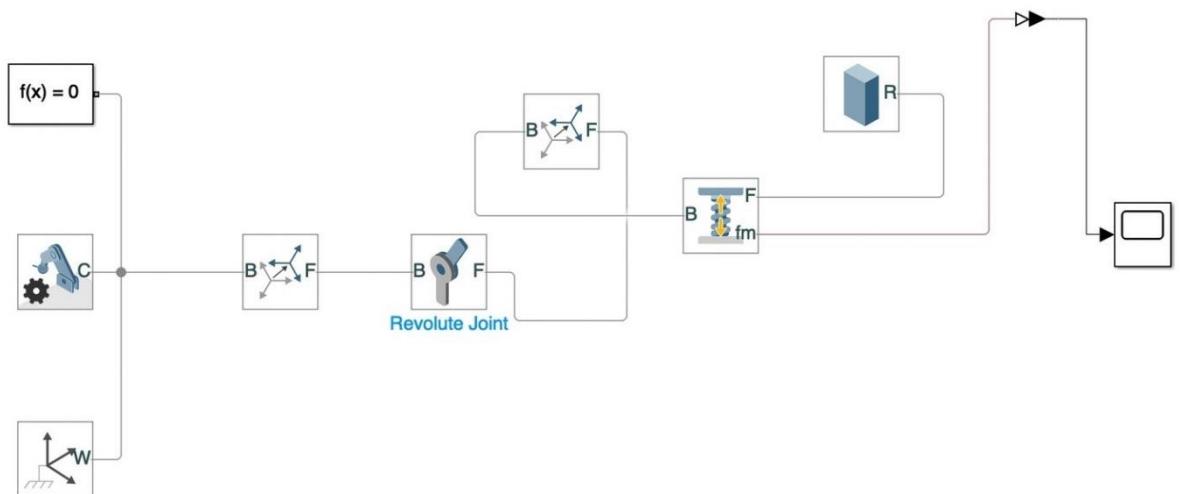


Рис. Схема построения заданной модели в Simulink

ЗАКЛЮЧЕНИЕ

В данной работе были изложены основные особенности проведения вычислений в среде MATLAB. Были приведены сведения об основных командах, операторах, функциях и процедурах MATLAB. Приведены примеры, которые поясняют и иллюстрируют работу по использованию процедур. Полезно для студентов в качестве начального ознакомления с системой MATLAB и приобретения навыков работы с ней.



Список использованной литературы

Учебный практикум

Лекция 1.

https://github.com/BMSTU-Automatic-Control-Systems-IU1-1/project-technological-practice-matlab/blob/main/12-09-2022-2022.09_Lecture_1.pdf

Лекция 2. Функции. Массивы. Структуры. Строки.

https://github.com/BMSTU-Automatic-Control-Systems-IU1-1/project-technological-practice-matlab/blob/main/12-09-2022-2022.09_Lecture_2.pdf

Лекция 3. Построение типовых графиков.

https://github.com/BMSTU-Automatic-Control-Systems-IU1-1/project-technological-practice-matlab/blob/main/12-09-2022-2022.09_Lecture_3.pdf

Введение в MatLAB - В.В.Шестаков

https://github.com/BMSTU-Automatic-Control-Systems-IU1-1/method-of-calculation/blob/main/14-09-2022-Metoda_MatLab.pdf

MathWorks MATLAB & Simulinc - А.Л. Масленников

<https://github.com/BMSTU-Automatic-Control-Systems-IU1-1/method-of-calculation/blob/main/14-09-2022-matlab.pdf>

Наместников С.М. Основы программирования в MatLab / Сборник лекций: УлГТУ, Ульяновск. - 2011. - 55 с.

Кочегурова Е.А. Особенности системы MatLAB для решения задач вычислительной математики: Томск. – 2013. – 110 с.

В.Д. Дубовец, В.А. Столер, В.Ф. Бондаренко. Построение графических моделей в среде MATLAB . Минск: БГУИР, 2014 – 70 с.

Дьяконов В. П.MATLAB. Полный самоучитель. – М.: ДМК Пресс, 2012 – 768 с.