

Лабораторная работа №1

Работа в режиме командной строки в системе MATLAB

Цель работы: Освоение принципов и правил работы в системе MATLAB в режиме командной строки. Знакомство с основными типами данных в MATLAB.

1. Краткая характеристика MATLAB

MATLAB (сокращение от англ. «MatrixLaboratory») – это пакет прикладных программ для решения задач технических вычислений и одноименный язык программирования, используемый в этом пакете.

Основными достоинствами системы MATLAB являются быстрые численные алгоритмы, визуализация вычислений, интерактивная среда программирования, связь с языками программирования C, C++, FORTRAN и Java, а так же обмен данными с другими приложениями.

Система MATLAB поддерживает работу в двух режимах:

- режим командной строки;
- режим программирования.

В режиме командной строки система MATLAB представляет собой мощный инженерный калькулятор, позволяющий пошагово реализовывать достаточно сложные алгоритмы расчетов.

В режиме командной строки диалог с пользователем осуществляется в окне Command Window, при этом должны соблюдаться следующие основные правила:

1. Ввод осуществляется после приглашения системы в виде последовательности символов '>>' в начале строки и завершается нажатием клавиши 'Enter', после чего на экран выводится результат вычисления введенного выражения;
2. Символ ';' (точка с запятой) в конце строки блокирует автоматический вывод результата вычисления;
3. Символ '...' (многоточие) в конце строки означает, что следующая строка является продолжением предыдущей;
4. Любой текст, введенный после символа '%' (процент), воспринимается как комментарий.

В текущей сессии вводимый построчно текст сохраняется в буфере, что позволяет вызывать его повторно с помощью клавиш управления курсором ↑ и ↓.

Последовательность команд текущей сессии отображается в окне CommandHistory и

сохраняется после выхода из MATLAB.

2. Интерфейс MATLAB

Пользователь имеет возможность самостоятельно настроить интерфейс системы MATLAB. При использовании настроек по умолчанию (Layout→Default) открывается интерфейс (рисунок 1), который содержит следующие элементы:

1. Командное окно (CommandWindow), предназначенное для работы в режиме командной строки;
2. Рабочая область (Workspace), в которой отображаются текущие переменные;
3. Путь к рабочему каталогу, используемому для хранения файлов данной сессии;
4. Содержимое рабочего каталога (CurrentFolder);
5. История команд (CommandHistory), введенных в командное окно.

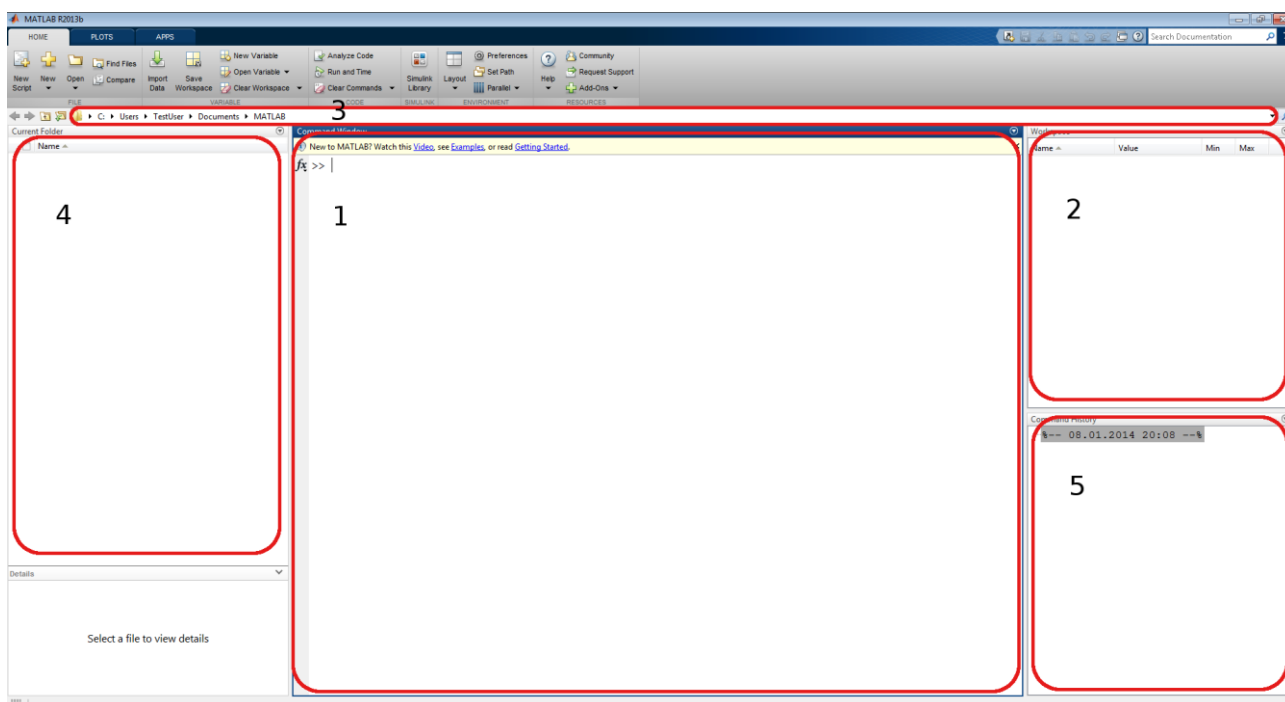


Рисунок 1. Основные элементы пользовательского интерфейса MATLAB

3. Работа в режиме командной строки

В режиме командной строки MATLAB выделяют следующие базовые объекты:

1. команды;
2. операторы;
3. константы;

4. переменные;
5. функции;
6. выражения.

Команды

Команда – это объект языка MATLAB со стандартным именем, предназначенный для взаимодействия с системой MATLAB и имеющий формат:

<команда><содержательная часть>

где:

<команда> – стандартное имя команды;

<содержательная часть> – уточняется для каждой конкретной команды и может отсутствовать.

Например, для очистки окна CommandWindowиспользуется команда `clc`:

```
>> clc
```

Операторы

Оператор – это объект языка MATLAB со стандартным именем, предназначенный для разработки программ.

В режиме командной строки используют оператор присваивания. Различают две его разновидности:

- оператор явного присваивания (=) с форматом <имя переменной>=<выражение>. С помощью оператора присваивания '=' переменной с указанным именем присваивается значение выражения .

- Оператор неявного присваивания с форматом <выражение>. В данном случае значение выражения присваивается переменной со стандартным именем `ans` .

Переменные

Переменная – это объект языка MATLAB, который в процессе вычисления может менять свое значение.

Переменные представляются своими именами – идентификаторами. Для создания переменной в MATLAB необходимо присвоить значение переменной с заданным именем с помощью оператора присваивания '='. Для примера создадим переменную `x=2`:

```
>> x = 2
x =
    2
```

Встречая имя новой переменной, MATLAB автоматически создает новую переменную и выделяет для нее необходимый объем памяти. Если данная переменная уже

существовала, то MATLAB изменит ее значение.

Имя переменной в MATLAB может содержать латинские буквы (A–Z и a–z), цифры (0–9), знак подчеркивания (_). Имя обязательно должно начинаться с буквы и не должно совпадать с ключевыми словами MATLAB. Так же необходимо учитывать, что MATLAB чувствителен к регистру, и поэтому будет считать переменные x и X разными. В таблице 1 приведены примеры допустимых и недопустимых имен переменных:

Таблица 1. Примеры имен переменных

| допустимые имена переменных | недопустимые имена переменных |
|-----------------------------|-------------------------------|
| x6 | 6x |
| lastValue | end |
| n_factorial | n! |

Тип созданной переменной не объявляется, а определяется типом присваиваемого значения. Основные типы данных, используемых в MATLAB, рассмотрены далее в п.п. 4. В данном примере переменная x будет иметь тип double (число с плавающей точкой двойной точности), так как все числа в MATLAB по умолчанию имеют тип double. Все переменные по умолчанию считаются матрицами.

Удаление созданных переменных

Все созданные переменные и их значения отображаются в окне workspace (рисунок 2).

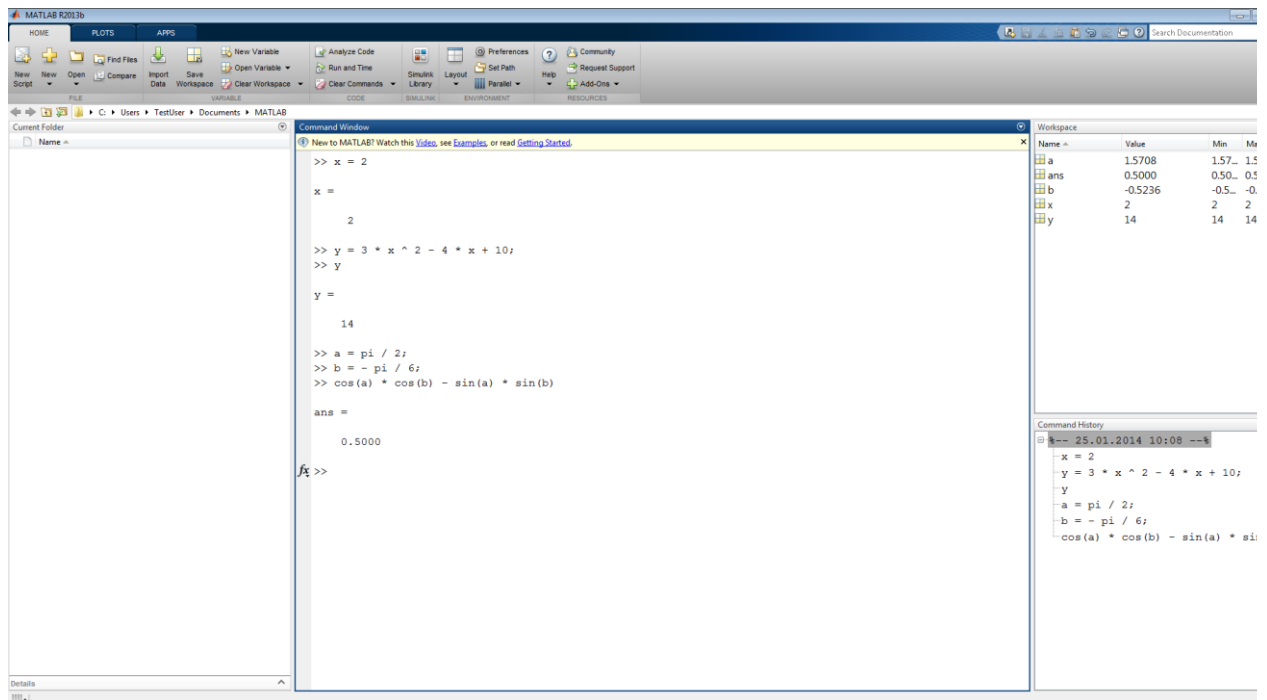


Рисунок 2. Текущие переменные

Для удаления переменных в MATLAB из рабочей области предусмотрена команда `clear`. Так, для удаления переменной `x` вводят:

```
>>clear x
```

Для удаления нескольких переменных достаточно после команды `clear` перечислить имена переменных через пробел. Если нужно удалить сразу все переменные, команду `clear` вводят без аргументов:

```
>>clear
```

Константы и системные переменные

Константа – это предварительно определенное числовое или символьное значение, представленное уникальным именем. Числа (например 1, -2 и 1.23) являются безымянными числовыми константами. Другие виды констант в MATLAB принято называть системными переменными, поскольку, с одной стороны, они задаются системой при ее загрузке, а с другой – могут переопределяться. Основные системные переменные, применяемые в системе MATLAB, указаны в таблице 2.

Таблица 2. Основные системные переменные

| Системная переменная | Значение |
|----------------------|---|
| <code>i, j</code> | мнимая единица (корень квадратный из -1: $0.0000 + 1.0000i$) |
| <code>pi</code> | число π (3.1415926...) |
| <code>eps</code> | погрешность операций над числами с плавающей точкой ($2.2204e-16$) |
| <code>realmin</code> | наименьшее число с плавающей точкой ($2.2251e-308$) |
| <code>realmax</code> | наибольшее число с плавающей точкой ($1.7977e+308$) |
| <code>inf</code> | значение машинной бесконечности; |
| <code>ans</code> | переменная, хранящая результат последней операции и обычно вызывающая его отображение на экране дисплея |
| <code>NaN</code> | указание на нечисловой характер данных (Not-a-Number) |
| <code>true</code> | логическая переменная, означающая «истину», в MATLAB ассоциирована с единицей: <code>true == 1</code> |
| <code>false</code> | логическая переменная, означающая «ложь», в MATLAB ассоциирована с нулем: <code>false == 0</code> |

Функции

В MATLAB различают две разновидности функций – встроенные и внешние.

Встроенная функция – это объект языка MATLAB со стандартным именем, предназначенный для выполнения действий с параметрами (аргументами), заключенными в круглые скобки. Обращение к функции происходит по ее имени и имеет формат:

$$[Y1, Y2, \dots] = \langle \text{имя функции} \rangle (X1, X2, \dots)$$

где:

$\langle \text{имя функции} \rangle$ – стандартное имя функции;

$(X1, X2, \dots)$ – входные параметры в круглых скобках;

$[Y1, Y2, \dots]$ – выходные параметры в квадратных скобках, если выходной параметр единственный, скобки можно не ставить.

Некоторые встроенные функции MATLAB приведены в приложении. Подробное описание встроенных функций и порядок их вызова доступен в командной строке с помощью команды `help <имя функции>`.

Внешние функции (функции пользователя) как правило должны быть расположены в рабочем каталоге.

Выражения

Выражение – это объект языка MATLAB, представляющий собой имеющую смысл совокупность констант, переменных и функций, объединенных символами операций. Особенностью выражений MATLAB является то, что входящие в них переменные по умолчанию считаются матрицами. Различаются следующие виды выражений в MATLAB:

- арифметические;
- логические;
- символьные.

В выражениях могут использоваться любые операции, а так же их комбинации. При этом операции в выражении последовательно, в соответствии с установленным для них приоритетом. Приоритет операций представлен в таблице 3.

Таблица 3. Приоритеты операций

| Уровень | Операции |
|---------|---|
| 1 | поэлементное транспонирование «.'», поэлементное возведение в степень «.^», эрмитово сопряженное транспонирование матрицы «'», возведение матрицы в степень «^» |
| 2 | унарное сложение «+», унарное вычитание «-», логическое отрицание «~» |
| 3 | поэлементное умножение «.*», поэлементное деление «./» и «.\», матричное умножение «*», матричное деление «/» и «\» |
| 4 | сложение «+», вычитание «-» |
| 5 | оператор формирования массивов «:» |
| 6 | Операторы отношения (>, >=, <, <=, ==, ~=) |
| 7 | Поэлементное «И» & |
| 8 | Поэлементное «ИЛИ» |
| 9 | Сокращенное «И» && |
| 10 | Сокращенное «ИЛИ» |

Внутри каждого уровня операторы имеют равный приоритет и вычисляются в порядке следования слева направо. Заданный по умолчанию порядок следования может быть изменен с помощью круглых скобок.

Приведем пример:

```
>> x = [0 pi*5/2];  
>> y = ( sin( 2 * x ) + 1 ) .* x
```

```
y =
```

```
0 7.8540
```

Простые вычисления

В командной строке в режиме диалога можно набрать команду или выражение и, нажав Enter, получить ответ (ans). Вычислим значение полинома $3x^2 - 4x + 10$ при значении $x = 2$. Для этого в командное окно введем:

```
>> 3 * 2 ^ 2 - 4 * 2 + 10
```

и нажмем Enter.

```
ans =
```

```
14
```

Теперь попробуем вычислить более сложное выражение $\cos(a)\cos(b) - \sin(a)\sin(b)$ при $a = \frac{\pi}{2}$, $b = -\frac{\pi}{6}$. Заметим, что для вычисления представленного выражения необходимо использовать встроенные функции MATLAB `sin`, `cos` и константу `pi`. Теперь вычислим данное выражение:

```
>>cos(pi/2)*cos(-pi/6)-sin(pi/2)*sin(-pi/6)
```

```
ans =
```

```
0.5000
```

Проведем вычисления с использованием переменных:

```
>> x = 2;
```

```
>> y = 3 * x ^ 2 - 4 * x + 10;
```

Отметим, что после выражения был поставлен символ «;». Символ «;» подавляет вывод в консоль значения вычисленного выражения или результата, присвоенного переменной. Для того, чтобы вывести значение данной переменной, достаточно просто ввести ее имя:

```
>> y
```

```
y =
```

```
14
```

Приведем другой пример:

```
>> a = pi / 2;
```

```
>> b = - pi / 6;
```

```
>>cos(a) * cos(b) - sin(a) * sin(b)
ans =
    0.5000
```

В приведенном выше примере результат вычисления выражения не был присвоен какой либо переменной. В таких случаях используется оператор неявного присваивания, то есть MATLAB присваивает результаты вычислений переменной `ans`.

4. Основные типы данных, используемых в MATLAB

Как говорилось ранее, тип созданной переменной в MATLAB определяется типом присваиваемого ей значения. Все типы данных в MATLAB являются матричными. Матрицей называется прямоугольная таблица, состоящая из m строк и n столбцов. Если число строк не равно числу столбцов ($m \neq n$), то матрица называется прямоугольной, если количество строк и столбцов одинаково, матрица называется квадратной. В случае, когда в прямоугольной матрице число строк $m = 1$, матрица представляет собой вектор-строку. Если же число столбцов $n = 1$ – матрица является вектором-столбцом. Система MATLAB воспринимает скаляр или 1 символ как вектор с единственным элементом, или точнее – матрицу размером 1×1 .

Матрица содержит элементы одного типа данных. В матрице обращение к элементу осуществляются по индексу. В круглых скобках указывается сначала номер строки, а затем, через запятую, номер столбца элемента, к которому выполняется обращение. Например, запись `A(1, 3)` означает, что выполняется обращение к элементу матрицы `A`, содержащемуся в 1 строке, 3 столбце.

Для каждого типа данных устанавливаются свои операции. Рассмотрим основные типы данных в MATLAB.

Числовые (NUMERIC) типы данных

Все арифметические операции в системе MATLAB выполняются над числами с плавающей запятой двойной точности - `double`. Любая переменная, перед выполнением операций над ней, должна быть преобразована к типу `double`. При создании числовая переменная по умолчанию получает тип `double`.

При вводе и выводе чисел в MATLAB можно использовать формат чисел с фиксированной запятой. Дробная часть числа отделяется от целой точкой, а не запятой.

Все операции в MATLAB реализуются посредством функций. Например, для выполнения операции сложения MATLAB вызывает функцию plus(). Перечень основных встроенных в систему MATLAB функций для числовых данных приведены в учебном пособии «Алгоритмизация и структурное программирование в MATLAB».

Для матриц в MATLAB существуют две различные операции, связанные с умножением, делением и возведением в степень. Для обозначения поэлементной операции перед соответствующим оператором ставится точка. Так, при матричном умножении двух матриц с помощью оператора «*» будет выполнено умножение согласно правилам линейной алгебры, порядок следования матриц в выражении определяет значение произведения. При поэлементном умножении двух матриц с помощью оператора «.*» каждый элемент первой матрицы будет умножен на соответствующий элемент второй матрицы. Порядок следования матриц в выражении не важен. Пример матричного и поэлементного умножения двух матриц приведен ниже.

```
>> a=[1 2; 3 4]
```

```
a =
```

```
    1    2  
    3    4
```

```
>> b=[4 3; 2 1]
```

```
b =
```

```
    4    3  
    2    1
```

```
>> c=a*b
```

```
c =
```

```
    8    5  
   20   13
```

```
>> c=b*a
```

```
c =
```

```
   13   20  
    5    8
```

```
>>d=a.*b
```

```
d =
```

```
    4    6  
    6    4
```

```
>>d=b.*a
```

```
d =
```

```
    4    6  
    6    4
```

Логический тип данных (LOGICAL)

Значения этого типа являются результатом вычислений условных и логических выражений и участвуют во всевозможных условных операторах языка. Логическое выражение может принимать одно из двух значений: «Истина» («true») или «Ложь» («false»). В MATLAB «Истина» ассоциируется с единицей, а «Ложь» – с нулем, хотя запись «true» и «false» также допустима.

Значения логического типа получаются как результат операций отношения («==», «~=», «<», «>», «<=», «>=») и логических операций (конъюнкция, дизъюнкция, отрицание, сумма по модулю 2), список которых с примерами использования в MATLAB приведен в учебном пособии «Алгоритмизация и структурное программирование в MATLAB».

В MATLAB существует ряд логических функций, начинающихся со слова «is» («быть», «являться» с англ.), с помощью которых переменные проверяются на различные условия. Например, функция `isscalar(x)` возвращает логическую 1, если переменная x является матрицей размером 1×1 (т.е. скаляром или символом), и логический 0 в противном случае.

Следует отметить, что система MATLAB воспринимает как «Ложь» только 0 (типа `double`), а все остальные числовые или строковые значения воспринимаются как «Истина», однако результатом логических вычислений могут быть только «0» или «1».

В отличие от некоторых других языков программирования, результат вычисления логического выражения может быть использован в арифметических выражениях. В качестве примера рассчитаем значение кусочно заданной функции с использованием операций отношения. Дана функция:

$$y = \begin{cases} \sin(x), & x < \pi; \\ \sqrt{(x - \pi)}, & x \geq \pi. \end{cases}$$

```
>> x=0:0.1:6; %создать вектор-строку x со значениями от 0 до 6 с шагом 0.6
>> y=(x<pi).*sin(x)+(x>=pi).*(x-pi).^0.5;
>>plot(x,y) % для визуализации результатов используется функция ...
построения двумерного графика plot()
```

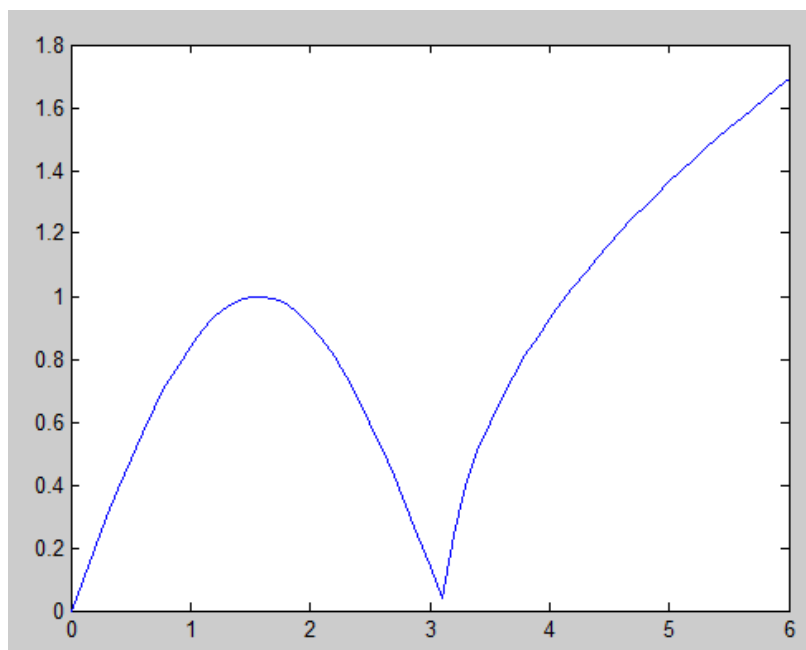


Рисунок 3. График кусочно заданной функции

В данном примере задается вектор аргументов x со значениями от 0 до 6 с шагом 0,1. После этого вычисляется вектор значений функции y . Логическое выражение $(x < \pi)$ будет равно нулю, если переменная x больше или равна π . В таком случае все первое слагаемое в функции будет равно 0, значимым будет лишь второе слагаемое. Аналогично для второго слагаемого, если логическое выражение $(x >= \pi)$ равно нулю, второе слагаемое также равно нулю и значимым является первое слагаемое.

Символьный тип данных

Символьный тип (`char`) – это тип данных, состоящих из одного символа (знака, буквы, кода). Значением типа `char` может быть любой символ из таблицы кодировки символов ASCII. Строки в MATLAB являются векторами, элементами которых являются символы, для них применимы многие матричные операции. Так, например, можно объединить строки с помощью символов «`[,]`» или «`[;]`». Система MATLAB позволяет создавать многострочные переменные символьного класса, являющиеся двумерными массивами символов.

Для создания строки в MATLAB необходимо ее значение (текст) заключить в одинарные кавычки. Приведем пример создания строки:

```
>>str = 'Hellow, world!'
str =
```

```
Hello, world!
```

Внутри одиночных кавычек можно помещать все доступные символы, например буквы, цифры, символы подчеркивания, двойные кавычки и др. Однако, так как строка сама ограничивается одинарными кавычками, то для того чтобы поместить в строку одинарную кавычку, необходимо поместить в нее две следующие подряд одинаковые кавычки.

```
>>str = 'You''re right!'
str =
You're right!
```

Система MATLAB обладает рядом функций для работы с символьными переменными и строками, например функции создания и проверки строк.

5. Основы работы с матрицами и векторами

Создание матриц и векторов

Как отмечалось ранее, переменные в MATLAB представляют собой матрицы. Рассмотрим способы ввода матрицы с указания полного списка элементов. В MATLAB матрицы задаются с помощью специальных символов («[]», «[,]», «[;]»), а так же с помощью стандартных функций (`horzcat()` – создает строку, `vertcat()` – создает столбец). При задании матриц необходимо следить за равенством длин строк, ее образующих. Для примера создадим числовую матрицу $A = \begin{bmatrix} 1.2 & 2.4 & 3.1 \\ 4.3 & 5.3 & 6.4 \end{bmatrix}$:

```
>> A = [1.2 2.4 3.1; 4.3 5.3 6.4]
A =
    1.2000    2.4000    3.1000
    4.3000    5.3000    6.4000
```

В данном случае элементы матрицы, находящиеся в одной строке, разделяют пробелами, а строки разделяются символом «;». Так же элементы, находящиеся в одной строке, можно разделить с помощью запятых:

```
>> A = [1.2, 2.4, 3.1; 4.3, 5.3, 6.4]
A =
    1.2000    2.4000    3.1000
    4.3000    5.3000    6.4000
```

Матрицы в MATLAB можно задавать не только с помощью вертикального объединения строк, как в предыдущем примере, но и при помощи горизонтального объединения столбцов:

```
>> A = [[1.2;4.3] [2.4;5.3] [3.1;6.4]]
A =
    1.2000    2.4000    3.1000
    4.3000    5.3000    6.4000
```

При обращении к элементу матрицы в MATLAB после имени матрицы следует указать в круглых скобках два индекса, разделенных запятой. Первый индекс обозначает номер строки, а второй – номер столбца. Нумерация строк и столбцов начинается с единицы.

```
>> A(2,3)
ans =
    6.4000
```

При обращении к строке (столбцу) матрицы необходимо указать номер строки в качестве первого (второго) индекса, а на месте второго (первого) индекса ввести символ двоеточие «:».

```
>> A(2,:)
ans =
    4.3000    5.3000    6.4000
```

Также MATLAB позволяет создать специальные матрицы, для чего используются встроенные функции MATLAB.

Векторы создаются аналогично матрицам:

```
>> x = [1 2 3]
x =
     1     2     3
>> y = [1; 2; 3]
y =
     1
     2
     3
>> A=vertcat(2,5,8); B=horzcat(5,3,1);
>>A
A =
```

```

2
    5
    8
>> B
B =
    5    3    1

```

В данном случае x и B – это векторы-строки, а u и A – векторы-столбцы. Отличия вектора-столбца от вектора-строки проявляются при использовании матричных операций и использовании функций.

Рассмотрим еще один важный способ задания вектора-строки – оператор двоеточия «:». Например:

```

>> 1:10

ans =

    1    2    3    4    5    6    7    8    9   10

```

В данном случае был создан вектор-строка со значениями от 1 до 10 с шагом 1 (задается по умолчанию). При необходимости шаг можно указать явно:

```

>> 1:2:10

ans =

    1    3    5    7    9

```

В данном случае значение 10 не содержится в векторе, так как оно задает лишь максимальное значение (или минимальное, при отрицательном шаге), которое будет содержаться в векторе.

Шаг так же может быть отрицательным:

```

>> 0 : -pi/4 : -2 * pi

ans =

    0.0000    -0.7854    -1.5708    -2.3562    -3.1416    -3.9270    -4.7124

```

Поэлементные операции над матрицами

Рассмотрим поэлементные операции над матрицами/векторами.

Большинство функций MATLAB могут работать с матрицами и векторами.

Рассмотрим в качестве примера функцию `sin`:

```
>>x = 0 :pi/6 : pi
x =
    0    0.5236    1.0472    1.5708    2.0944    2.6180
3.1416
>>sin(x)
ans =
0    0.5000    0.8660    1.0000    0.8660    0.5000    0.0000
```

Результат данных вычислений является вектор такого же размера, что и входной аргумент функции `sin`, содержащий значения синуса элементов вектора `x`.

Так же можно производить поэлементные операции сложения, вычитания, умножения, деления и возведения в степень для матриц/векторов одинаковой размерности.

Например:

```
>> x = [1 2 3];
>> y = [3 2 1];
>> x + y
ans =
    4    4    4
>> x - y
ans =
   -2    0    2
>>x .* y
ans =
    3    4    3
>>x ./ y
ans =
    0.3333    1.0000    3.0000
>>x .^ y
ans =
    1    4    3
```

Однако, при поэлементном умножении матриц разного размера, получим ошибку:

```
>> x = [1 2 3];
>> y = [3; 2; 1];
>>x .* y
Error using .*
```

Matrixdimensionsmustagree.

Отметим, что для поэлементного умножения, деления и возведения в степень необходимо перед операцией ставить точку: «.*», «./», «.^», иначе будут применяться матричные операции. При вычислении более сложных операций в MATLAB нужно помнить о приоритете этих операций (см. таблицу 3).

6. Преобразование типов данных в MATLAB

Как отмечалось ранее, основными типами данных в MATLAB являются числовые, логические и символьные. Зачастую, при написании программы необходимо преобразовать данные из одного типа в другой. Например, для того, чтобы подписать график функции текст подписи должен представлять собой строковую переменную. Данное из числовой переменной `a` с помощью функции `num2str` преобразуется в строку `b`.

```
>> a=sin(pi/2)+2*cos(pi/4)

a =

    2.4142

>> b=num2str(a)

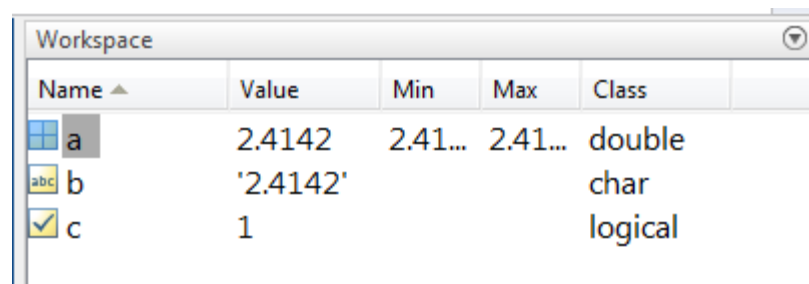
b =

    2.4142
>> c=(a>0)




c =

    1
```

В рабочей области было создано три переменные – `a` типа `double`, `b` типа `char` и `c` типа `logical`.



| Name | Value | Min | Max | Class |
|------|----------|---------|---------|---------|
| a | 2.4142 | 2.41... | 2.41... | double |
| b | '2.4142' | | | char |
| c | 1 | | | logical |

Следует обратить внимание, что для обозначения каждого из перечисленных типов данных в рабочей области MATLAB существует свой значок (иконка):  - тип double,  - тип char,  - тип logical.

Так же удобно использовать функции преобразования типов данных для проверки типов переменных. В частности, при организации диалога с пользователем удобно использовать преобразование типов для оценки корректности ввода данных. По результату преобразования можно судить о том, содержит ли строка только цифры или так же и символы. Например, функция `str2double(s)` (см. таблицу 4 приложения) преобразует строку `s` в число (скаляр) типа `double`. Если строка `s` содержала символы, математические выражения или представляла собой вектор или матрицу, функция `str2double(s)` вернет `Nan` (Not-a-number). Следует так же отметить, что при преобразовании с помощью функции `str2double` символ запятая `,` опускается.

```
>> a='23.2';
>> str2double(a)
ans =
    23.2000

>> b='hello';
>> str2double(b)
ans =
    NaN

>> c='12,4';
>> str2double(c)
ans =
    124
```

Так же MATLAB обладает возможностями вычисления математических выражений, содержащихся в символьной переменной или строке. В частности, это реализуется с помощью функции `eval()`.

```
>> str='sin(pi/2)';
>> eval(str)
ans =
    1
```

```
>> x = 0 : 0.5 : pi;
>> str = '( sin( 2 * x ) + 1 ) .* ( x .^ 2 )';
>> y=eval (str)
y =
    0    0.4604    1.9093    2.5675    0.9728    0.2567    6.4853
```

7. Основы построения графиков в MATLAB

MATLAB обладает мощным инструментом визуализации результатов вычислений.

Построим график функции $y = (\sin(2x) + 1) \cdot x^2$ в MATLAB. Для этого создадим вектор-строку значений переменной от 0 до 6π с шагом 0,05.

```
>>x = 0 : 0.05 : 6*pi;
>>y = ( sin( 2 * x ) + 1 ) .* ( x .^ 2 );
>>plot(x, y);
>>xlabelx
>>ylabely
```

После ввода данных команд появится график представленных на рисунке 4.

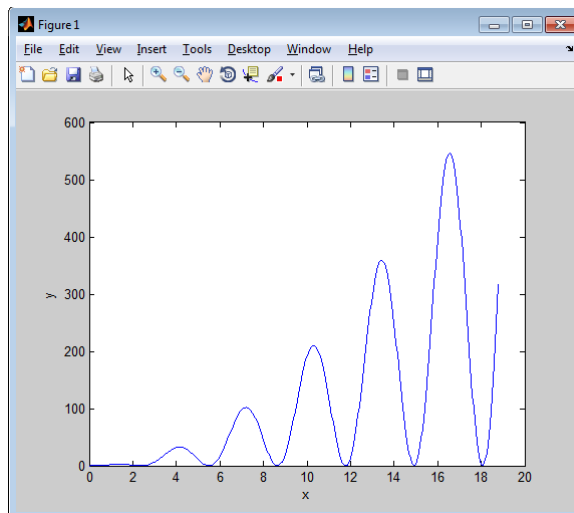


Рисунок 4. График в MATLAB

В простейшем случае для построения графиков используется функция `plot`, принимающая два входных аргумента, которыми являются векторы одинаковой длины, задающие координаты точек для построения графика. В первом аргументе содержится значение координат точек по оси абсцисс, а во втором – по оси ординат.

Для того, чтобы подписать оси графиков, используются команды `xlabel` и `ylabel`, за которыми через пробел следует название оси графика.

8. Справочная система в MATLAB (Help)

Подробную информацию об объектах и функциях MATLAB можно получить, используя справочную систему. Для вывода справки в консоль используют команду `help`. Для вывода справки по конкретному объекту используют `help<имя объекта>`. При этом в консоль выводится краткая справка по объекту, приводятся гиперссылки на близкие по значению или назначению объекты, а так же на HTML-страницу с расширенной справкой по объекту. В расширенной справке так же приводятся примеры использования рассматриваемого объекта.

Например, для получения справки о функции `sin`, введем в командной строке «`helpsin`»:

```
>> help sin
sin      Sine of argument in radians.
sin(X) is the sine of the elements of X.

See also asin, sind.

Overloaded methods:
codistributed/sin
gpuArray/sin
sym/sin

Reference page in Help browser
doc sin
```

Так же в MATLAB существует справка по ключевому слову, реализуемая командой «`lookfor<Ключевое слово>`».

9. Задания на лабораторную работу №1

Задание №1

1. Выбрать функцию для вычисления в соответствии с номером варианта (таблица 4);
2. Задать точку x_1 согласно номеру варианта;
3. Вычислить значение функции в точке x_1 и присвоить результат переменной y_1 ;
4. С использованием оператора двоеточия сформировать вектор x со значениями от x_{Min} до x_{Max} с шагом dx ;
5. Для каждого элемента вектора x вычислить спомощью значение функции, заданной по варианту, и записать результат в переменную y ;
6. Используя созданные вектора, построить график функции и подписать оси.

Задание №2

Используя операторы отношения, вычислить значения кусочно заданной функции по 100 равномерно распределенным точкам, охватывающими *все ее области определения* (3 отрезка) согласно номеру варианта (таблица 5), рассчитанные значения присвоить переменной y_2 и нарисовать график функции.

Задание №3

1. Создать строковые переменные: `str1`, в которую записать свою фамилию, переменную `str2`, в которую записать свое имя и переменную `str3`, в которую записать свое отчество;
2. С помощью объединения строк создать переменные `str4` и `str5`, содержащее ФИО, записанные в одну строку, и ФИО, записанные в три строки соответственно;
3. Найти, в каких позициях ФИО содержится буква «а»;
4. Определить, содержится ли в ФИО буква «ю»;
5. В переменной `str4` поменять все строчные буквы на прописные;
6. В переменной `str5` поменять все заглавные буквы на строчные.

Задание №4

1. Создать строковые переменные, содержащие данные согласно номеру варианта (таблица 6)
2. Преобразовать строковые переменные в числовые с помощью функции `str2double`. Объяснить полученные результаты.

3. Преобразовать строковые переменные в числовые с помощью функции `str2num`. Объяснить полученные результаты.
4. Применить к строковым переменным функцию `eval`. Объяснить полученные результаты.

Задание №5*(дополнительное)

Четырьмя способами сформировать матрицу, содержащую 2 строки и 3 столбца, элементами которой являются первые 6 значений вектора y 2 значений функции (задание 2). Названия матриц должны соответствовать A_i , где $i=1..4$. (Задание матриц с использованием « , » и « » для элементов в одной строке считается 1 способом). С помощью логических функций проверить матрицы на равенство, матрицу A_3 возвести в квадрат, поэлементно умножить матрицы A_1 и A_2 .

Таблица 4. Варианты задания №1

| Номер варианта | ЗАДАНИЕ №1 | | | | |
|-------------------|---|-----------------|---------|------------------|--------|
| | Функция для задания №1 | x1 | xMin | dx | xMax |
| I | II | III | IV | V | VI |
| 1 | $y = \sin(2x)^2 - \cos(5x) + 2$ | $\frac{\pi}{3}$ | -2π | $\frac{\pi}{25}$ | 2π |
| 2 | $y = \frac{\sin(x)^2 + \cos(-3x)^2}{x} + 2$ | $\frac{\pi}{3}$ | -2π | $\frac{\pi}{25}$ | 2π |
| 3 | $y = -5\cos(x^2) + x - \sin(x) \cdot x \cdot 2$ | $\frac{\pi}{3}$ | -2π | $\frac{\pi}{25}$ | 2π |
| 4 | $y = \frac{\operatorname{tg}(2x) \cdot \sin(\frac{x}{2})}{x^2 + 1}$ | $\frac{\pi}{3}$ | -2π | $\frac{\pi}{25}$ | 2π |
| 5 | $y = 6\sin(\frac{x}{2} + 1) + \cos(\frac{x^2}{10} - 3x)$ | $\frac{\pi}{3}$ | -2π | $\frac{\pi}{25}$ | 2π |
| 6 | $y = \frac{x^3 + 4x - 10}{7x^2 - 4x + 1}$ | 5 | -8 | 0.16 | 8 |
| 7 | $y = \frac{-x^2 + \frac{4}{5}x + 2}{2x^3 - 9x + 2}$ | 5 | -8 | 0.16 | 8 |
| 8 | $y = \frac{\frac{x^4}{2} - 10x^2 + 4}{-6x^3 - 2x + 2}$ | 3 | 0 | 0.08 | 8 |
| 9 | $y = \frac{x^2 - 4x + 4}{3x^2 - 4x + 4}$ | 3 | 0 | 0.08 | 8 |

| Номер варианта | ЗАДАНИЕ №1 | | | | |
|-------------------|--|-----------------|------------------|------------------|-----------------|
| | Функция для задания №1 | x1 | xMin | dx | xMax |
| I | II | III | IV | V | VI |
| 10 | $y = \frac{x^2 + 4x - 10}{2x^3 - 4x}$ | 3 | 0 | 0.08 | 8 |
| 11 | $y = \frac{e^{\frac{x}{4}}}{x} + \sin(x + 1) \cdot x$ | 4 | 1 | 0.08 | 9 |
| 12 | $y = 2e^{-(x-3)^2} + 2e^{-(x-6)^2}$ | 4 | 1 | 0.08 | 9 |
| 13 | $y = -x \cdot e^{-(x-4)^2} + x \cdot e^{-(x-9)^2}$ | 4 | 1 | 0.08 | 9 |
| 14 | $y = \ln\left(\frac{x}{4}\right) \cdot x \cdot \sin(x) + e^{-(x-4)^2}$ | 2 | 4 | 0.02 | 6 |
| 15 | $y = \frac{\ln\left(\frac{x}{5}\right) \cdot x + 10}{\ln\left(\frac{x}{5}\right)^2 \cdot x + \frac{1}{2}}$ | 2 | 4 | 0.02 | 6 |
| 16 | $y = \cos x + \sqrt{ \sin 2x } + \frac{4\sin^2 \frac{x}{2}}{\operatorname{tg} x}$ | $\frac{\pi}{3}$ | 0 | $\frac{\pi}{25}$ | 2π |
| 17 | $y = \left(\frac{2\sqrt{2^x + 3^{-x}} - 1}{\sin x + \cos(2x - 5)} \right)^3$ | 3 | 0 | 0.08 | 8 |
| 18 | $y = 3\sin^2\left(\frac{2x+5}{3x}\right) + \sqrt{\frac{ \cos 3x }{1+e^x}}$ | $\frac{\pi}{3}$ | $-\frac{\pi}{2}$ | $\frac{\pi}{25}$ | $\frac{\pi}{2}$ |
| 19 | $y = \frac{\cos(3x^2 + 2)}{8^x + 7^{-x}} + e^{(3x+2)}$ | $\frac{\pi}{4}$ | -2π | $\frac{\pi}{20}$ | 2π |
| 20 | $y = 2x + \cos \sqrt{\frac{8(x + 2x^2)}{3x}}$ | 3 | 1 | 0.08 | 9 |
| 21 | $y = 3\sin^2\left(\frac{2x+5}{3x}\right) + \sqrt{\frac{ \cos 3x }{1+e^x}}$ | $\frac{\pi}{3}$ | 0 | $\frac{\pi}{50}$ | $\frac{\pi}{2}$ |
| 22 | $y = \frac{-x^2 + \frac{4}{5}x + 2}{2x^3 - 9x + 2}$ | 5 | -8 | 0.16 | 8 |
| 23 | $y = \frac{\sin(x)^2 + \cos(-3x)^2}{x} + 2$ | $\frac{\pi}{3}$ | -2π | $\frac{\pi}{25}$ | 2π |
| 24 | $y = 2e^{-(x-3)^2} + 2e^{-(x-6)^2}$ | 4 | 1 | 0.08 | 9 |

| Номер варианта | ЗАДАНИЕ №1 | | | | |
|-------------------|---|-----------------|---------|------------------|--------|
| | Функция для задания №1 | x1 | xMin | dx | xMax |
| I | II | III | IV | V | VI |
| 25 | $y = \frac{x^2 - 4x + 4}{3x^2 - 4x + 4}$ | 3 | 0 | 0.08 | 8 |
| 26 | $y = \cos x + \sqrt{ \sin 2x } + \frac{4\sin^2 \frac{x}{2}}{\operatorname{tg} x}$ | $\frac{\pi}{3}$ | 0 | $\frac{\pi}{50}$ | π |
| 27 | $y = \frac{\operatorname{tg}(2x) \cdot \sin(\frac{x}{2})}{x^2 + 1}$ | $\frac{\pi}{3}$ | -2π | $\frac{\pi}{25}$ | 2π |

Таблица 5. Варианты задания №2

| Номер варианта | ЗАДАНИЕ №2 |
|-------------------|---|
| | Функция для задания №2 |
| I | VII |
| 1 | $y = \begin{cases} 3, & x \leq -4; \\ x^2 - 4 x + 3 , & -4 < x \leq 4; \\ 3 - (x - 4)^2, & x > 4. \end{cases}$ |
| 2 | $y = \begin{cases} 8 - (x + 6)^2, & x \leq -6; \\ x^2 - 6 x + 8 , & -6 < x < 5; \\ 3, & x \geq 5. \end{cases}$ |
| 3 | $y = \begin{cases} x^2 + 1, & x \leq 0; \\ 1 + 2x, & 0 < x < 2; \\ x - 2, & x \geq 2. \end{cases}$ |
| 4 | $y = \begin{cases} 0, & x < 0; \\ \operatorname{tg} x, & 0 \leq x < \frac{\pi}{4}; \\ 1, & x \geq \frac{\pi}{4}. \end{cases}$ |

| Номер варианта | <p style="text-align: center;">ЗАДАНИЕ №2</p> <p style="text-align: center;">Функция для задания №2</p> |
|-------------------|--|
| I | VII |
| 5 | $y = \begin{cases} \pi \sin x, & -2\pi \leq x < -\pi; \\ \pi - x , & -\pi \leq x < \pi; \\ \pi \sin^3 x, & \pi \leq x \leq 2\pi. \end{cases}$ |
| 6 | $y = \begin{cases} \frac{4}{x}, & x < -2; \\ 0,5x - 1, & -2 \leq x < 2; \\ -x^2 + 4x - 4, & x \geq 2. \end{cases}$ |
| 7 | $y = \begin{cases} \frac{x^2}{15}, & x < 12; \\ x - 20, & 12 \leq x < 25; \\ 3 \sin x, & x \geq 25. \end{cases}$ |
| 8 | $y = \begin{cases} e^{-0,4x+0,4\pi}, & x < \pi; \\ \sin(0,5x), & \pi \leq x < 4\pi; \\ \sqrt{(x-4\pi)}, & x \geq 4\pi. \end{cases}$ |
| 9 | $y = \begin{cases} e^{0,2x-0,4\pi}, & x < 2\pi; \\ \cos x, & 2\pi \leq x < 4\pi; \\ \ln(e-x+4\pi), & x \geq 4\pi. \end{cases}$ |
| 10 | $y = \begin{cases} 3, & x \leq -4; \\ x^2 - 4 x + 3 , & -4 < x \leq 4; \\ 3 - (x-4)^2, & x > 4. \end{cases}$ |
| 11 | $y = \begin{cases} 8 - (x+6)^2, & x \leq -6; \\ x^2 - 6 x + 8 , & -6 < x < 5; \\ 3, & x \geq 5. \end{cases}$ |
| 12 | $y = \begin{cases} x^2 + 1, & x \leq 0; \\ 1 + 2x, & 0 < x < 2; \\ x - 2, & x \geq 2. \end{cases}$ |

| Номер варианта | <p style="text-align: center;">ЗАДАНИЕ №2</p> <p style="text-align: center;">Функция для задания №2</p> |
|-------------------|--|
| I | VII |
| 13 | $y = \begin{cases} 0, & x < 0; \\ \operatorname{tg} x, & 0 \leq x < \frac{\pi}{4}; \\ 1, & x \geq \frac{\pi}{4}. \end{cases}$ |
| 14 | $y = \begin{cases} \pi \sin x, & -2\pi \leq x < -\pi; \\ \pi - x , & -\pi \leq x < \pi; \\ \pi \sin^3 x, & \pi \leq x \leq 2\pi. \end{cases}$ |
| 15 | $y = \begin{cases} \frac{4}{x}, & x < -2; \\ 0,5x - 1, & -2 \leq x < 2; \\ -x^2 + 4x - 4, & x \geq 2. \end{cases}$ |
| 16 | $y = \begin{cases} \frac{x^2}{15}, & x < 12; \\ x - 20, & 12 \leq x < 25; \\ 3 \sin x, & x \geq 25. \end{cases}$ |
| 17 | $y = \begin{cases} e^{-0,4x+0,4\pi}, & x < \pi; \\ \sin(0,5x), & \pi \leq x < 4\pi; \\ \sqrt{(x-4\pi)}, & x \geq 4\pi. \end{cases}$ |
| 18 | $y = \begin{cases} e^{0,2x-0,4\pi}, & x < 2\pi; \\ \cos x, & 2\pi \leq x < 4\pi; \\ \ln(e-x+4\pi), & x \geq 4\pi. \end{cases}$ |
| 19 | $y = \begin{cases} 3, & x \leq -4; \\ x^2 - 4 x + 3 , & -4 < x \leq 4; \\ 3 - (x-4)^2, & x > 4. \end{cases}$ |

| Номер варианта | ЗАДАНИЕ №2 Функция для задания №2 |
|-------------------|--|
| I | VII |
| 20 | $y = \begin{cases} 8 - (x + 6)^2, & x \leq -6; \\ x^2 - 6 x + 8 , & -6 < x < 5; \\ 3, & x \geq 5. \end{cases}$ |
| 21 | $y = \begin{cases} \pi \sin x, & -2\pi \leq x < -\pi; \\ \pi - x , & -\pi \leq x < \pi; \\ \pi \sin^3 x, & \pi \leq x \leq 2\pi. \end{cases}$ |
| 22 | $y = \begin{cases} 3, & x \leq -4; \\ x^2 - 4 x + 3 , & -4 < x \leq 4; \\ 3 - (x - 4)^2, & x > 4. \end{cases}$ |
| 23 | $y = \begin{cases} \frac{x^2}{15}, & x < 12; \\ x - 20, & 12 \leq x < 25; \\ 3 \sin x, & x \geq 25. \end{cases}$ |
| 24 | $y = \begin{cases} e^{0.2x - 0.4\pi}, & x < 2\pi; \\ \cos x, & 2\pi \leq x < 4\pi; \\ \ln(e - x + 4\pi), & x \geq 4\pi. \end{cases}$ |
| 25 | $y = \begin{cases} x^2 + 1, & x \leq 0; \\ 1 + 2x, & 0 < x < 2; \\ x - 2, & x \geq 2. \end{cases}$ |
| 26 | $y = \begin{cases} 0, & x < 0; \\ \operatorname{tg} x, & 0 \leq x < \frac{\pi}{4}; \\ 1, & x \geq \frac{\pi}{4}. \end{cases}$ |

| Номер варианта | ЗАДАНИЕ №2 | |
|-------------------|---|--|
| | Функция для задания №2 | |
| I | VII | |
| 27 | $y = \begin{cases} \frac{x^2}{15}, & x < 12; \\ x - 20, & 12 \leq x < 25; \\ 3\sin x, & x \geq 25. \end{cases}$ | |

Таблица 6. Варианты задания №4

| Номер варианта | Переменные | | | |
|-------------------|-----------------------|--------|-----------------|--------------------|
| | a | b | c | d |
| 1 | '1 2 3 4' | '8.87' | '2*sin(pi/4)+8' | 'help' |
| 2 | '1,2,3,4' | '3,1' | '1.2*5+2' | 'clc' |
| 3 | '1;2;3;4' | '35.6' | '3*cos(pi/3)' | 'clear all' |
| 4 | '1 2;2 2;3 2' | '5.89' | '8/150+0.24' | 'which' |
| 5 | '[2 3 8 9]' | '7,67' | '4*tan(pi/8)-1' | 'error' |
| 6 | '[2,3,8,9]' | '1.01' | '2*sin(pi/4)+8' | 'why' |
| 7 | '[2;3;8;9]' | '23.4' | '4*tan(pi/8)-1' | 'what' |
| 8 | '[3 2;1 4;3 1]' | '8,8' | '2*sin(pi/4)+8' | 'exist' |
| 9 | '[[1;4] [2;5] [3;6]]' | '0' | '2*sin(pi/4)+8' | 'realmax' |
| 10 | '[3;4] [9;7.7] [5;0]' | '0.45' | '1.2*5+2' | 'hello, world!' |
| 11 | '5 6 7 8' | '0,42' | '3*cos(pi/3)' | 'realmin' |
| 12 | '5,6,7,8' | '3.55' | '3*cos(pi/3)' | 'clc' |
| 13 | '5;6;7;8' | '11' | '8/150+0.24' | 'date' |
| 14 | '5 6;6 7;7 8' | '16' | '4*tan(pi/8)-1' | 'clock' |
| 15 | '[9 8 7 6]' | '10,2' | '2*sin(pi/4)+8' | 'zeros' |
| 16 | '9,8,7,6' | '2.3' | '1.2*5+2' | 'calendar' |
| 17 | '[9;8;7;6]' | '12,2' | '3*cos(pi/3)' | 'eye' |

| Номер варианта | Переменные | | | |
|-------------------|-----------------------|--------|-----------------|--------------------|
| | a | b | c | d |
| 18 | '[1 0;4 6;4 1]' | '2,1' | '8/150+0.24' | 'false' |
| 19 | '[[7;4] [3;7] [0;8]]' | '5,5' | '4*tan(pi/8)-1' | 'true' |
| 20 | '[2;3] [7;6] [8;9]' | '0' | '2*sin(pi/4)+8' | 'a+b' |
| 21 | '5;6;7;8' | '0.45' | '3*cos(pi/3)' | 'home' |
| 22 | '5 6;6 7;7 8' | '0,42' | '8/150+0.24' | 'ones' |
| 23 | '[9 8 7 6]' | '3.55' | '4*tan(pi/8)-1' | 'who' |
| 24 | '9,8,7,6' | '11' | '2*sin(pi/4)+8' | 'help help' |
| 25 | '[9;8;7;6]' | '1,6' | '4*tan(pi/8)-1' | 'why' |
| 26 | '1 0;4 6;4 1' | '100' | '2*sin(pi/4)+8' | 'now' |
| 27 | '[[7;4] [3;7] [0;8]]' | '2,3' | '2*sin(pi/4)+8' | 'close' |
| 28 | '[2;3] [7;6] [8;9]' | '12,2' | '3*cos(pi/3)' | 'hello, world!' |

10. Контрольные вопросы

1. Дайте определение «переменной». Что происходит при операции присвоения? Как обозначается оператор присвоения в MATLAB?
2. Какие существуют правила именования переменных в MATLAB? Приведите примеры допустимых и недопустимых имен.
3. Какой тип по умолчанию имеют все числа в MATLAB? Что представляет собой этот тип данных?
4. Какой оператор подавляет вывод в консоль значения вычисленного выражения?
5. Для чего предназначены операции `clc` и `clear`?
6. Что такое скалярное значение, матрица и вектор в MATLAB?
7. Что такое приоритет операций?
8. Чем отличается символ от строки?
9. Что означает `Nan` в MATLAB.
10. В чем отличие работы функций `str2double`, `str2num`, `eval` в MATLAB?

11. Требования к содержанию отчета

Отчет по лабораторной работе оформляется в любом текстовом редакторе и предоставляется в электронном виде. Отчет должен состоять из следующих разделов:

1. Титульный лист. На титульном листе необходимо указать номер и название лабораторной работы, номер варианта, ФИО и группу исполнителя, ФИО преподавателя.
2. Цель работы.
3. Задание на лабораторную работу в соответствии с номером варианта.
4. Ход работы, где указывается последовательность команд для командной строки согласно заданию и результат их исполнения.
5. Графики функций.
6. Выводы по работе.

К отчету прилагается файл с текстом отчета, который необходимо разместить в личном кабинете.