

Лабораторная работа 1

Работа в командном окне и в редакторе скриптов

Системные переменные

eps – относительная точность чисел с плавающей запятой (по умолчанию 2.2204e-016)

realmax – наибольшее положительное вещественное число (1.7977e+308)

realmin – наименьшее положительное вещественное число (2.2251e-308)

intmax – наибольшее положительное целое число (2147483647)

intmin – наименьшее положительное целое число (-2147483648)

ans – системная переменная, содержащая результат последней операции (в отсутствие результирующей переменной)

pi – число π (3.141592653589793 в формате long)

inf – бесконечно большое

NaN – нечисловой формат (not a number)

exp(1) – число e (2.718281828459046), нет зарезервированной системной переменной

end – последний элемент вектора или матрицы

Основные команды

% Комментарий

“;” после команды подавляет вывод результата на экран

clc % очистка командного окна Command Window

clear % очистить все переменные из рабочего пространства Workspace

clear x y % очистить переменные x и y

who % выводит список всех переменных в рабочем пространстве

whos % выводит список всех переменных в рабочем пространстве с их размером и типом данных

whos x y % выводит информацию о переменных x и y

Справочная система

- 1) Открытие справки в отдельном окне – вывод длинной справки с синтаксисом команды, возможными входными и выходными параметрами, примерами использования



- 2) Вывод короткой справки в командном окне с помощью команды **help whos** % выводит справку о команде whos

help elfun % выводит справку (список) элементарных математических функций (тригонометрические, экспоненциальные (логарифмы, возведение в степень), комплексные, функции округления)

help matfun % выводит справку (список) матричных функций

help specfun % выводит справку (список) специальных математических функций

Указание к выполнению заданий. Выполните предлагаемый код в Matlab, в комментариях возле каждой строки напишите результат выполнения команд, что делает каждая команда.

Простейшие операции

Задание 1. Выполните вычисления и объясните результат

clc, clear

1+2, 3-4, 3*5, 10/3, 3^3

format rational

1/3

format short

1/3, 1/3.

1/0 % какой результат?

0/0 % какой результат?

Задание 2

Задайте векторы:

d1=[1 1 1], d2=[2 2 2]

d3=[5; 5; 5], d4=d2'

Выполните операции, объясните результат. Укажите, в каком случае выдается ошибка и объясните, почему она выдается, прокомментируйте эту строку.

d1+d2

d3+d4

d1+d3

d1*d2

d4*d2

d1/d2 % посмотрите help на команду правого деления / (mrdivide)

d1./d2

Задание 3

Выполните операции, объясните результат (при этом результаты заданий 1-2 остаются в сессии)

a='MatLab forever'

who

whos % почему в **a** такое количество байт?

clear d2

whos

clear d1, d3 % объясните результат

who

clear

Конструкторы матриц и операции с векторами и матрицами

Задание 4. Матрицы из нулей (конструктор **zeros**). Выполните команды и объясните результат

```
zeros(5)
zeros(2,3)
zeros(4,3,5)
A=[1,2,3;4,5,6]
B= zeros(size(A))
```

Задание 5. Матрицы из единиц (конструктор **ones**) и **единичная матрица** (конструктор **eye**). Выполните команды и объясните результат

```
ones(5)
ones(2,3)
ones(4,3,5)
A=[1,2,3;4,5,6]
eye(5)
B=eye(size(A)) %как выглядит единичная матрица в этом случае?
```

Задание 6. Случайные матрицы. Выполните команды и объясните результат

```
N=rand % случайное число из равномерного распределения на открытом интервале (0,1)
n=3; m=4; %выдается ли на экран результат?
A = rand(n,m) % псевдослучайная матрица nxm со значениями из стандартного равномерного распределения в открытом интервале (0,1)
A1=rand(5)
A2=rand(2,3)
```

randn(4,2) % матрица 4x2, случайные числа генерируются с помощью нормального распределения нулевым средним и единичной дисперсией

```
N= randi(20) % псевдослучайное целое число в диапазоне от 1 до 20
A= randi([-5 7],3,4) % случайная матрица 3 x 4 из целых чисел в диапазоне от -5 до 7
B = randi(10,5) % случайная матрица 5x5 из целых чисел в диапазоне от 1 до 10
A3=randi([3 9],3,3)
```

Функции **randi** нет в версии Matlab R2008a. Вместо нее для генерации целочисленной случайной матрицы размера $m \times n$ с элементами в диапазоне $[a, b]$ (от a до b включительно) можно использовать комбинацию команд **round** и **rand**:

```
round(rand(m,n)*(b-a)+a)
```

Например, вместо **randi([3 9],3,3)** нужно записать **round(rand(3)*6+3)**

Задание 7.1. Выбор элементов матрицы, строк и столбцов. Выполните команды и объясните результат

A=[1 2 3; 4 5 6; 7 8 9]

B=[1 1 1; 2 2 2; 3 3 3]

A(2,3)

A(4)

A(:)

A(2,:)+B(3,:)

B(:,1)+A(:,2:3)

Задание 7.2. Выбор верхне- и нижнетреугольной части матрицы, диагонали.

Выполните команды и объясните результат

A=rand(6)

AL=tril(A)

AL1= tril(A,-2)

AU= triu(A)

AU1=triu(A,2)

AD=diag(A), v=diag(A,3)

B=diag(v)

Задание 7.3. Повороты матриц

A=rand(3), B=rand(4)

Alr=fliplr(A), Blr=fliplr(B) %как работает разворот влево-вправо?

Aud=flipud(A), Bud=flipud(B) %как работает разворот вверх-вниз?

Ar=rot90(A) %как работает разворот на 90 градусов?

Задание 8. Операции с матрицами. Выполните команды и объясните результат

A=randi(10,3), B=randi([-5 5],3)

A+B, A-B, A*B

% операции с точкой выполняются поэлементно

A.*B % чем отличается от A*B?

A.*10, A*10 % есть ли различие?

A.^2

A^2

A=A+i

A' % транспонирование ли матрицы?

A.' % чем отличается от предыдущего?

inv(B) % обратная матрица к B

B^(-1) % обратная матрица к B

sum(A), sum(A') % объясните, как работает функция sum

sum(A,1)

sum(A,2)

sum(diag(A))

sum(diag(fliplr(A)))

sum(sum(A))

prod(A), prod(A') % объясните, как работает функция prod

```
prod(A,1)
prod(A,2)
max(A(:)), max(max(A)) % объясните, как работает функция max
max(A)
max(A,[],1)
max(A,[],2)
```

Операции сравнения и логические операции

Задание 9. Выполните следующие операции сравнения и объясните результат

```
123+456==579
3>9
[3 7 4]>[3 2 3] % поэлементное сравнение
t=ans
all(t) % если в t все логические единицы, то результат true (или 1)
any(t) % если в t есть хотя бы одна логические единицы, то результат true (или 1)
```

```
2<2.5000
7*8>=56
2^32<=65536
```

```
X=[1 2 3]
Y=['1','2','3']
Z=['1','2','A']
whos X Y Z
double(Y), double(Z)
X==Y % Как происходит сравнение чисел и символов?
X<Y
Z==Y
Z>=Y
```

5

Задание 10. Логические операции с массивами. Задайте массивы и выполните логические операции, объясните результат

```
P=[1 1 1]
Q=[0 1 0]
R=[0 0 1]
```

```
Q|R
P&R
~Q
```

Задание 11. Операции над множествами. Придумайте самостоятельно примеры операций над множествами, используйте в качестве множеств векторы и примените к ним следующие функции Matlab: **unique**, **union**, **intersect**, **setxor**, **ismember**.

Задание 12. Контроль типов данных

Замечание. Язык нестрогой типизации обуславливает необходимость постоянного, насколько это требует логика программы, контроля типов.

Чтобы найти соответствующие функции контроля типов в справке Help следует набрать **is*** в строке поиска. В этом случае вы получите список всех функций контроля типов:

isequal – истина, если аргументы - матрицы равны поэлементно

isscalar – истина, если аргумент скаляр

isvector – истина, если аргумент вектор

isrow – истина, если аргумент строка

iscolumn - истина, если аргумент столбец

ismatrix – истина, если аргумент матрица

Выполните следующие операции и объясните результат

x=[1 2 3], y=x'

A=rand(2), B=A*A^(-1), C=eye(2)

isrow(x), isrow(y)

isvector(x), isvector(y)

iscolumn(x), iscolumn(y)

ismatrix(A), ismatrix(x), ismatrix(y)

ismatrix(2.56), isscalar(2/3)

isequal(B,C) % Результат истина или ложь? Почему такой результат?

Задание 13. Конкатенация матриц и удаление элементов

Задайте случайные матрицы r1 и r2 размера 3x3, выполните конкатенацию по вертикали и по горизонтали двумя способами: с помощью квадратных скобок и функции **cat**

Rr=[r1;r2] %дополнительно вычислите Rr с помощью функции **cat**

Rc=[r1 r2] %дополнительно вычислите Rc с помощью функции **cat**

Rr(:,end) = [] % удаление последнего столбца

Rc(2,:)=[]

Rr(1:2:end)=0 % каждый элемент матрицы с нечетным индексом обнуляется

Rc(2:2:end)=0

Задание 14. Разреженные матрицы (матрицы, которые содержат большое количество нулевых элементов)

Выполните следующие операции и объясните результат

n=10;

As=sprand(n,n, 0.05) % случайная разреженная матрица размера nхn плотности 5%

spy(As) % графическое представление структурспреженной матрицы (расположение ненулевых элементов)

Af=full(As)

B = bucky;

spy(B), nnz(B)

Задание 15. Элементарные математические функции (комплексные и функции округления)

Задайте комплексное число, найдите его характеристики, объясните результат

```
z=2-3j
abs(z)
real(z)
imag(z)
conj(z) % комплексно-сопряженное, сравните с z'
z'
isequal(conj(z),z')
```

Задайте вектор вещественных x и по результатам сформулируйте правила работы функций округления и получения остатка от деления (используйте справочную систему).

```
x=[-2.5, 3.2, 4.8, 3/4]
ceil(x)
fix(x)
floor(x)
round(x)
n1=23, n2=-23, m = 5
mod(n1,m), rem(n1,m) % чем отличаются mod и rem?
mod(n2,m), rem(n2,m)
```

Задание 16. Задание последовательностей (векторов-строк)

```
q=2:2:8 % начальное значение: шаг: конечное значение
r=3:3:12
v=q+r
dotv_1=sum(v.^2) % результат – скалярное произведение (v,v)
dotv_2=dot(v,v)
dots_3=v*v % уберите ошибку, представив третий способ вычисления скалярного
произведения (v,v) с помощью матричного умножения
dotqr_1=dot(q,r) % добавьте еще два способа вычисления скалярного произведения
(q,r)
```

Пользовательские функции в Matlab

Пользовательская функция с именем `func_name` может быть определена либо в основном файле, либо в отдельном `m`-файле с именем `func_name.m`. Отдельный файл-функция должен находиться в той же папке, что и файл с вызовом функции. Где располагать описание функции – зависит от версии Matlab.

- Во всех версиях Matlab определения функций могут располагаться в отдельных `m`-файлах. При этом для Matlab 2008 — это единственный способ описания функций, их нельзя задать в основном скрипте.

При описании функций в основном файле:

- В версиях Matlab до R2024a все определения функций должны располагаться после их вызова функции и всего остального кода, описания функций располагается в отдельной секции (после section break).
- В Matlab R2024a и более поздних версиях все определения функций могут располагаться как до, так и после вызова функции.

Функция с одним входным и одним выходным параметром

Пример. Определить функцию, которая вычисляет среднее значение элементов вектора

```
function y = average(x)
if ~isvector(x)
    error('Input must be a vector')
end
y = sum(x)/length(x);
end

%вызов функции
z = 1:99;
average(z)
```

Функция с несколькими входными и выходными параметрами

Пример. Определить функцию, которая максимальное и минимальное значение среди всех элементов двух матриц или векторов.

```
function [m,M] = minmax(A,B)
C=[A(:);B(:)];
m=min(C);
M=max(C);
end

% вызов функции
A=rand(2),B=rand(3,1)
[minAB,maxAB] = minmax(A,B)
```

Задание 17. Процедура-функция без параметров

Создайте функцию без параметров для вычисления площади треугольника по формуле Герона $s = \sqrt{p(p-a)(p-b)(p-c)}$, здесь p – полупериметр; a, b, c – длины сторон треугольника. Функция должна запрашивать параметры интерактивно по команде **input** (**a=input('введите a')** и т.п.) Протестируйте работу функции.

Задание 18. Функции с одним входным параметром

Создайте две функции, которые вычисляют:

1. Максимальное значение элементов главной диагонали матрицы.
2. Сумму всех элементов матрицы, вне главной диагонали.

Задание 19. Функция с несколькими входными и выходными параметрами

Создайте функцию, которая вычисляет корни квадратного уравнения x_1, x_2 , где квадратное уравнение задается своими коэффициентами a, b, c .