

Информационные технологии

Управленческая пирамида предприятия



Стратегический уровень – аналитические и прогнозирующие системы (OLAP)

Функциональный уровень – MES-, ERP-системы

Операционный уровень – АСУ ТП

АСУ ТП



3 – SCADA-системы

2 – управление
ВВОДОМ/ВЫВОДОМ

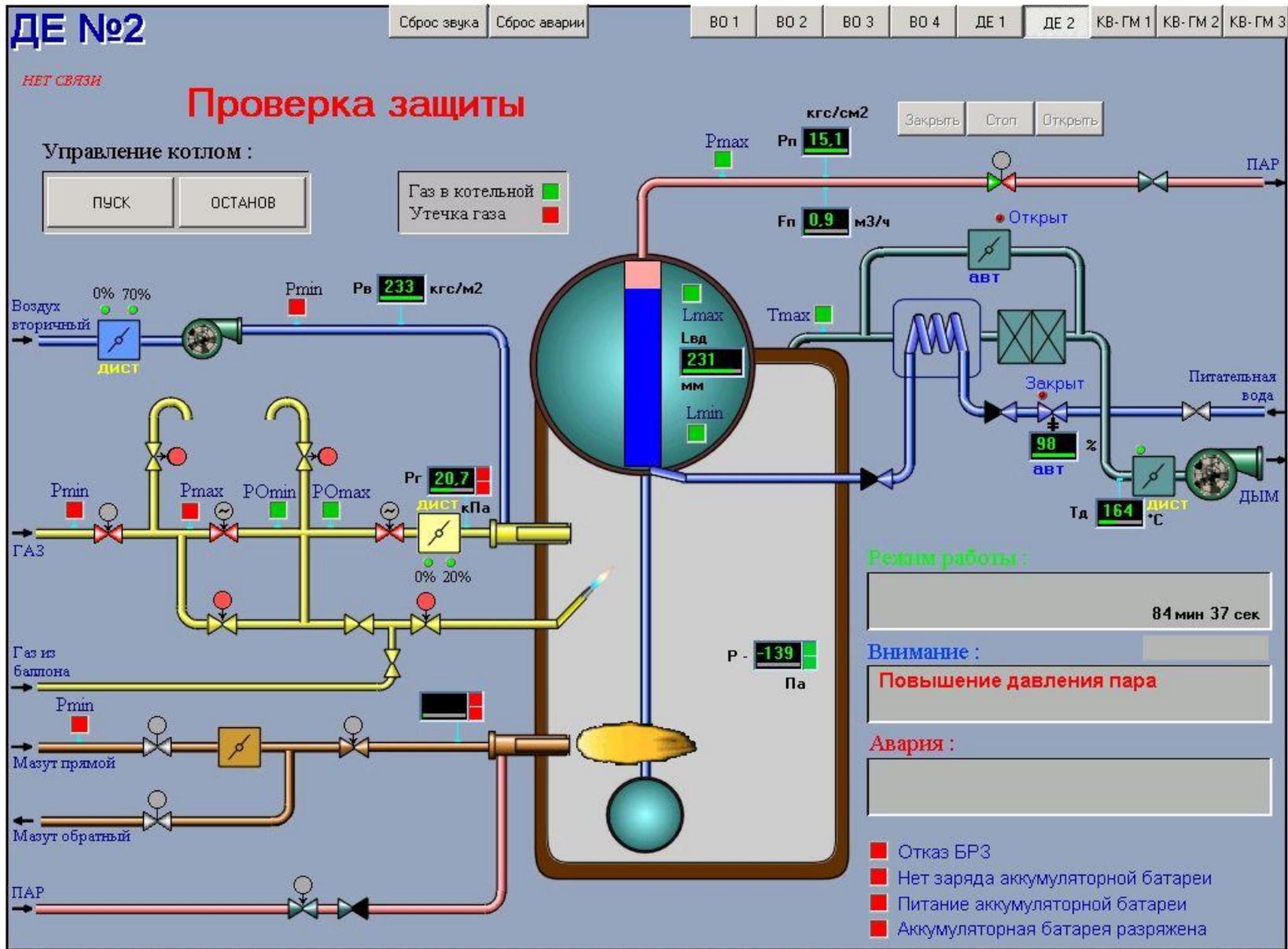
1 – ВВОД/ВЫВОД

SCADA-СИСТЕМЫ

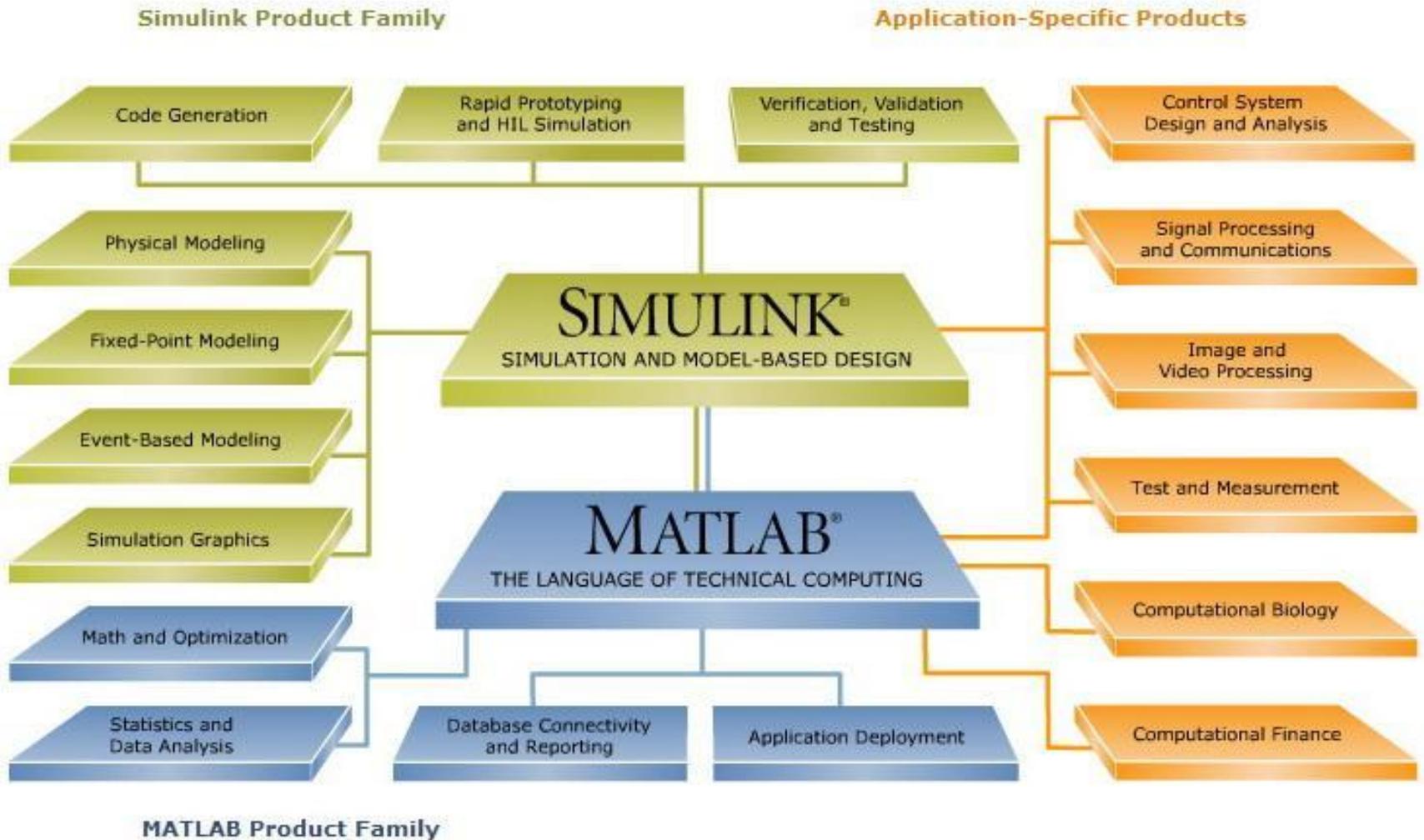
Решаемые задачи:

- обмен данными с УСО;
- обработка информации в реальном времени;
- логическое управление;
- отображение информации;
- ведение архивов;
- сигнализация;
- подготовка и генерирование отчетов и тп.

SCADA-СИСТЕМЫ



MATLAB



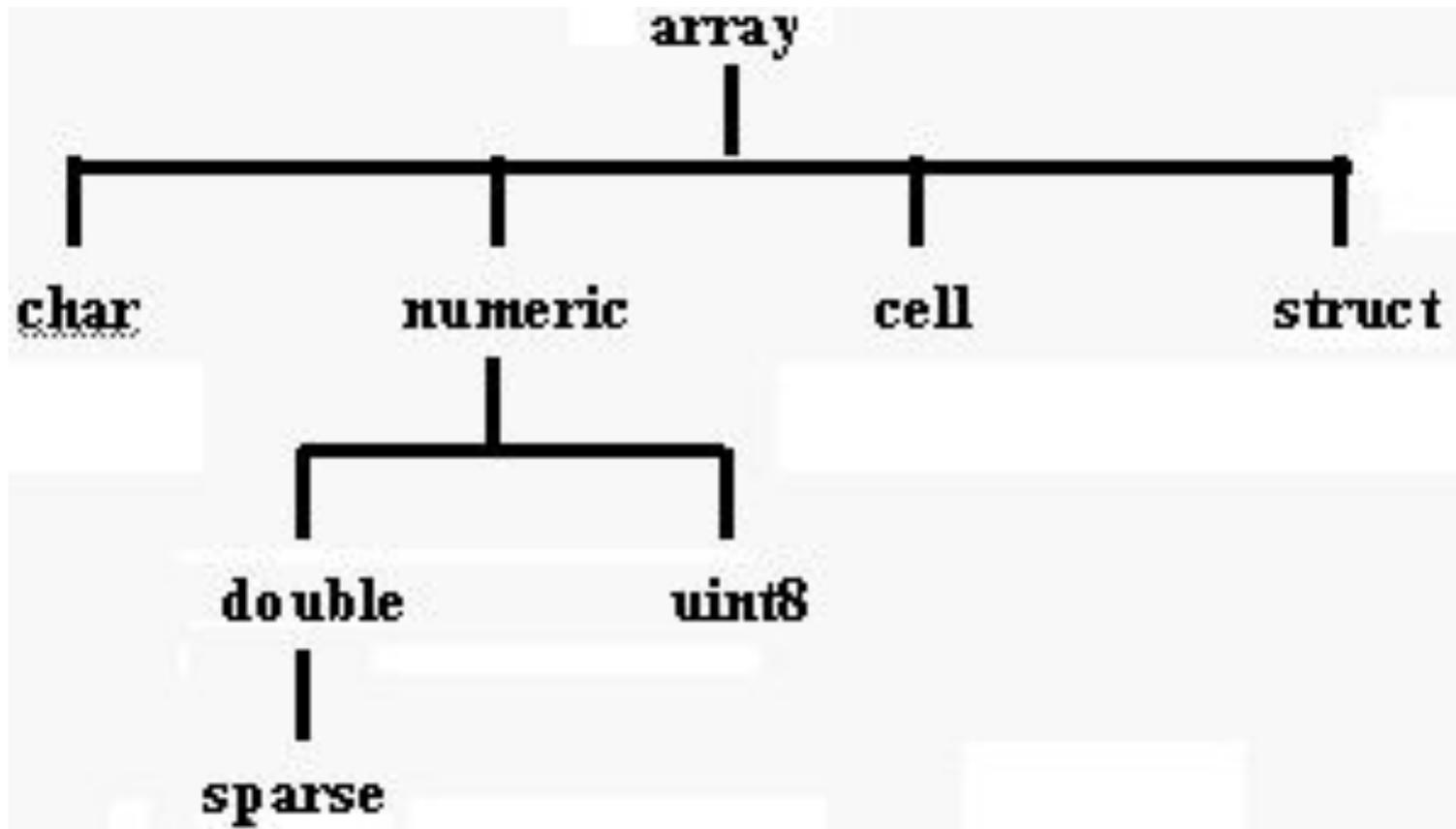
ПЕРЕМЕННЫЕ В МАТЛАВ

- переменные не требуют объявления
- любая операция присваивания создает переменную
- имя переменной начинается с латинской буквы, далее буквы, цифры подчеркивания
- имя переменной не должно превышать 31 символ
- различаются символы верхнего и нижнего регистра

СПЕЦИАЛЬНЫЕ ПЕРЕМЕННЫЕ

ans	Последний результат; если выходная переменная не указана, то MATLAB использует переменную ans.
eps	Точность вычислений с плавающей точкой; определяется длиной мантиссы и для PC $\text{eps} = 2.220446049250313\text{e-}016$
realmax	Максимальное число с плавающей точкой, представимое в компьютере; для PC $\text{realmax} = 1.797693134862316\text{e+}308$.
realmin	Минимальное число с плавающей точкой, представимое в компьютере; для PC $\text{realmin} = 2.225073858507202\text{e-}308$.
pi	Специальная переменная для числа π : $\text{pi} = 3.141592653589793\text{e+}000$.
i, j	Специальные переменные для обозначения мнимой единицы
inf	Специальная переменная для обозначения символа бесконечности
NaN	Специальная переменная для обозначения неопределенного значения - результата операций типа: $0/0$, inf/inf .

БАЗОВЫЕ ТИПЫ ДАННЫХ



БАЗОВЫЕ ТИПЫ ДАННЫХ

Тип	Пример	Описание
Double	$[1\ 2; 3\ 4]$ $5 + 6i$	Числовой массив удвоенной точности (это наиболее распространенный тип переменной в системе MATLAB)
Char	'Привет'	Массив символов (каждый символ - длиной 16 битов), часто именуется строкой.
Cell	{ 17 'привет' eye (2)}	Массив ячеек. Элементы этого массива содержат другие массивы. Массивы ячеек позволяют объединить связанные данные, возможно различных размеров, в единую структуру.
Struct	A.day = 12; A.color = 'Red'; A.mat = magic(3);	Массив записей. Он включает имена полей. Поля сами могут содержать массивы. Подобно массивам ячеек, массивы записей объединяют связанные данные и информацию о них.

БАЗОВЫЕ ТИПЫ ДАННЫХ

Разряженные матрицы

A =

7	0	0	0	0	0	8	0	0	0
10	0	0	0	0	0	3	0	0	0
7	0	0	0	0	0	2	0	0	0
4	0	0	0	0	0	3	0	0	0
8	0	0	0	0	0	4	0	0	0
1	0	0	0	0	0	5	0	0	0
1	0	0	0	0	0	6	0	0	0
1	0	0	0	0	0	4	0	0	0
2	0	0	0	0	0	4	0	0	0
3	0	0	0	0	0	5	0	0	0

Объем в памяти 800 байт
8 байт на каждый элемент

B=sparse(A)

B =

(1,1)	7
(2,1)	10
(3,1)	7
(4,1)	4
(5,1)	8
(6,1)	1
(7,1)	1
(8,1)	1
(9,1)	2
(10,1)	3
(1,7)	8
(2,7)	3
(3,7)	2
(4,7)	3
(5,7)	4
(6,7)	5
(7,7)	6
(8,7)	4
(9,7)	4
(10,7)	5

Объем в памяти 568 байт !

РАБОТА С МАТРИЦАМИ

Команда	Результат	Примечание
$a = [1\ 2\ 3\ 4]$	$a = (1, 2, 3, 4)$	Задаёт строку
$b = [1; 2; 3; 4]$	$b = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$	Задаёт столбец
$A = [1\ 2; 3\ 4]$	$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$	Квадратная матрица
$a:h:b$ $1:3:5$ $1:-2:-5$ $1:-2:5$	$(a, a + h, a + 2h, \dots, b)$ $(1, 4)$ $(1, -1, -3, -5)$ Empty matrix: 1-by-0	Арифметическая прогрессия

ФУНКЦИИ ГЕНЕРИРОВАНИЯ МАТРИЦ

Функция	Примечание
<code>zeros(m,n)</code>	Нулевая матрица
<code>ones(m,n)</code>	Матрица состоящая из единиц
<code>eye(m,n)</code>	Единичная матрица
<code>rand(m,n)</code>	Матрица со случайными элементами, равномерно распределенными на отрезке [0,1]

Обращение к элементам матрицы

```
% зададим матрицу
```

```
a = [ 1  2  3  4  5  6  7; ...  
      8  9 10 11 12 13 14; ...  
     15 16 17 18 19 20 21; ...  
     22 23 24 25 26 27 28; ...  
     29 30 31 32 33 34 35; ...  
     36 37 38 39 40 41 42; ...  
     43 44 45 46 47 48 49]
```

a =

1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	32	33	34	35
36	37	38	39	40	41	42
43	44	45	46	47	48	49

```
% запишем в b первый и второй столбец a  
b = a(:,1:2)
```

b =

1	2
8	9
15	16
22	23
29	30
36	37
43	44

Обращение к элементам матрицы

```
% выделим главную диагональ матрицы a
```

```
c = diag(a)
```

c =

1

9

17

25

33

41

49

```
% выделим побочной диагонали матрицы a
```

```
d = diag(fliplr(a))
```

d =

7

13

19

25

31

37

43

Обращение к элементам матрицы

```
% обнулим все элементы кроме главной диагонали  
c = a.*eye(size(a))
```

c =

```
1  0  0  0  0  0  0  
0  9  0  0  0  0  0  
0  0 17  0  0  0  0  
0  0  0 25  0  0  0  
0  0  0  0 33  0  0  
0  0  0  0  0 41  0  
0  0  0  0  0  0 49
```

ПРИОРИТЕТ ВЫПОЛНЕНИЯ АРИФМЕТИЧЕСКИХ ОПЕРАТОРОВ

уровень 1:	поэлементное транспонирование (\prime), поэлементное возведение в степень (\wedge), эрмитово сопряженное транспонирование матрицы (\prime), возведение матрицы в степень (\wedge);
уровень 2:	унарное сложение (+), унарное вычитание (-);
уровень 3:	умножение массивов (\cdot^*), правое деление ($\cdot /$), левое деление массивов ($\cdot \backslash$), умножение матриц ($*$), решение систем линейных уравнений - операция ($/$), операция (\backslash);
уровень 4:	сложение (+), вычитание (-);
уровень 5:	оператор формирования массивов (:).

ОСНОВНЫЕ МАТЕМАТИЧЕСКИЕ ФУНКЦИИ

Функция	Примечание
$\text{sqrt}(x)$	вычисление квадратного корня
$\text{exp}(x)$	возведение в степень числа e
$\text{pow2}(x)$	возведение в степень числа 2
$\text{log}(x)$	вычисление натурального логарифма
$\text{log10}(x)$	вычисление десятичного логарифма
$\text{log2}(x)$	вычисление логарифма по основанию 2
$\text{sin}(x)$	синус угла x , заданного в радианах
$\text{cos}(x)$	косинус угла x , заданного в радианах
$\text{tan}(x)$	тангенс угла x , заданного в радианах
$\text{cot}(x)$	котангенс угла x , заданного в радианах
$\text{asin}(x)$	арксинус
$\text{acos}(x)$	арккосинус
$\text{atan}(x)$	арктангенс

АРИФМЕТИЧЕСКИ ОПЕРАТОРЫ

$$A = [1 \ 2; 3 \ 4];$$

$$B = 10;$$

$$C = A - B$$

$$C =$$

$$\begin{array}{cc} -9 & -8 \end{array}$$

$$\begin{array}{cc} -7 & -6 \end{array}$$

$$A = [1 \ 2; 3 \ 4]$$

$$B = [4 \ 3; 2 \ 1]$$

$$C = A - B$$

$$C =$$

$$\begin{array}{cc} -3 & -1 \end{array}$$

$$\begin{array}{cc} 1 & 3 \end{array}$$

АРИФМЕТИЧЕСКИ ОПЕРАТОРЫ

A = [3 9 5];

B = [2 1 5];

C = A./B. ^ 2

C = 0.7500 9.0000 0.2000

C = (A./B). ^ 2

C = 2.2500 81.0000 1.0000

b = sqrt (A(2)) + 2*B (1)

b = 7

АРИФМЕТИЧЕСКИ ОПЕРАТОРЫ

$$A = [1 \ 2; 3 \ 4];$$

$$B = A^2;$$

$$D = A.^2;$$

$$B = [7 \ 10; 15 \ 22];$$

$$D = [1 \ 4; 9 \ 16];$$

АРИФМЕТИЧЕСКИ ОПЕРАТОРЫ

$A = [1 \ 2; 3 \ 4];$

$B = \text{sum}(A);$

$B =$

4 6

$C = \text{sum}(\text{sum}(A));$

$C =$

10

АРИФМЕТИЧЕСКИ ОПЕРАТОРЫ

```
A = [1 2; 3 4];
```

```
B = 3;
```

```
C = A + B;
```

```
C =
```

```
4 5
```

```
6 7
```

```
C = A * B;
```

```
C =
```

```
3 6
```

```
9 12
```

```
C = A .* B;
```

```
C =
```

```
3 6
```

```
9 12
```

```
C = A / B;
```

```
C =
```

```
0.3333 0.6667
```

```
1.0000 1.3333
```

```
C = A ./ B;
```

```
C =
```

```
0.3333 0.6667
```

```
1.0000 1.3333
```

Поиск максимального значения

`M = max(A)`

`M = max(A,[],dim)`

`M = max(A,[],nanflag)`

`M = max(A,[],dim,nanflag)`

`[M,I] = max(____)`

`M = max(A,[],'all')`

`M = max(A,[],vecdim)`

`M = max(A,[],'all',nanflag)`

`M = max(A,[],vecdim,nanflag)`

`[M,I] = max(A,[],____,'linear')`

`C = max(A,B)`

`C = max(A,B,nanflag)`

Поиск максимального значения

$A = [1\ 5\ 3\ 4; 5\ 6\ 7\ 8; 9\ 4\ 11\ 2]$

1	5	3	4
5	6	7	8
9	4	11	2

$B = \max(A);$

9	6	11	8
---	---	----	---

$B = \max(A,[],1);$

9	6	11	8
---	---	----	---

$B = \max(A,[],2);$

5
8
11

Поиск максимального значения

Максимальное значение среди всех элементов

матрицы?
 $A = [1\ 5\ 3\ 4; 5\ 6\ 7\ 8; 9\ 4\ 11\ 2]$

```
1  5  3  4
5  6  7  8
9  4 11  2
```

```
B = max(max(A)); % так себе
```

```
11
```

```
B = max(A(:)); % норм
```

```
11
```

```
A(:)'
```

```
1  5  9  5  6  4  3  7 11  4  8  2
```

Поиск максимального значения

Координаты максимального значения?

$A = [1\ 5\ 3\ 4; 5\ 6\ 7\ 8; 9\ 4\ 11\ 2]$

1	5	3	4
5	6	7	8
9	4	11	2

$[M, I] = \max(A)$

$M =$

9	6	11	8
---	---	----	---

$I =$

3	2	3	2
---	---	---	---

Статистические функции

corrcoef	Correlation coefficients
cov	Covariance matrix
max	Largest elements in array
cummax	Cumulative maximum
mean	Average or mean value of array
median	Median value of array
min	Smallest elements in array
cummin	Cumulative minimum
mode	Most frequent values in array
std	Standard deviation
var	Variance

Построение графиков

plot(y)

plot(x, y)

plot(x, y, s)

plot(x1, y1, s1, x2, y2, s2, ...)

Маркер Тип линии

- непрерывная
- штриховая
- : пунктирная
- . штрих-пунктирная

Маркер Тип точки

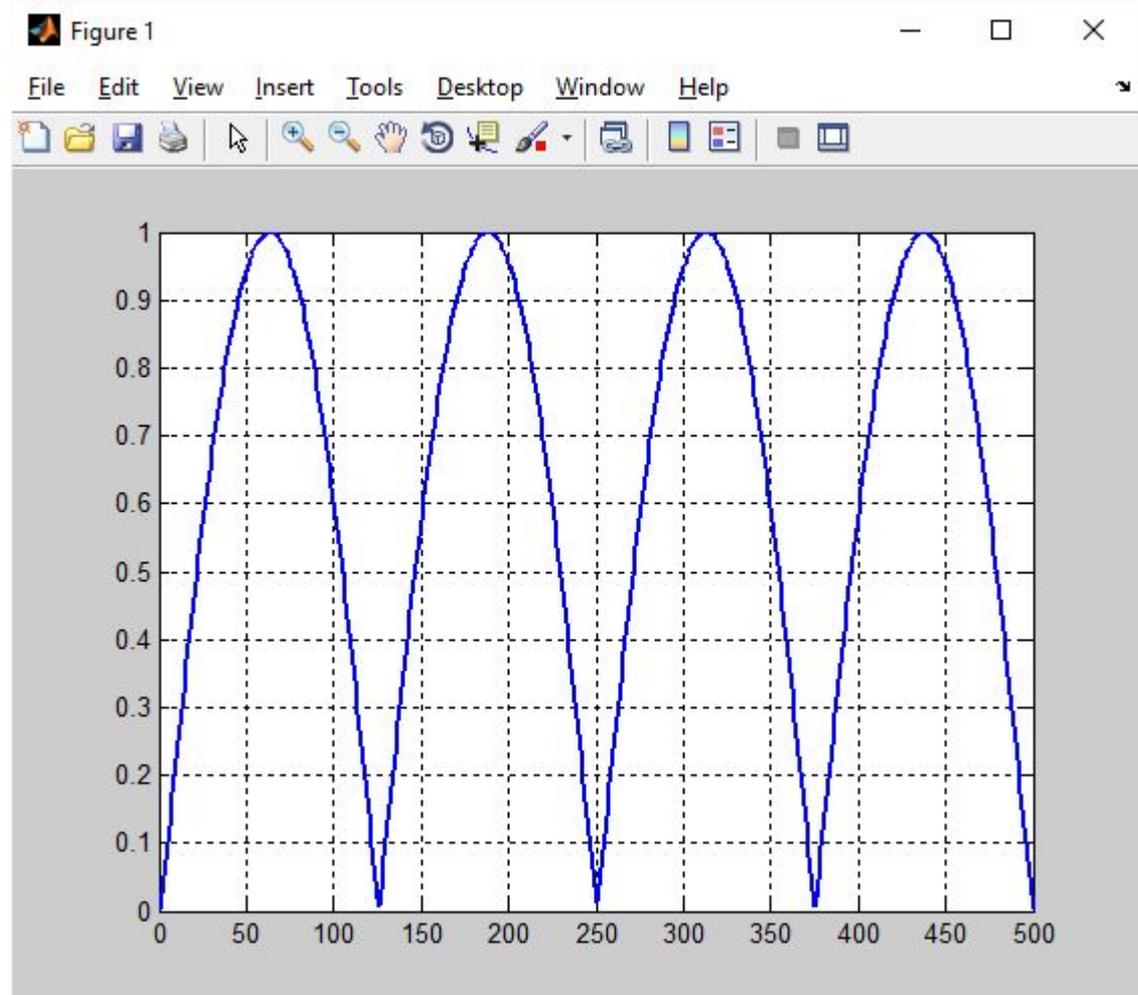
- . точка
- + плюс
- * звёздочка
- o кружок
- x крестик

Маркер Цвет линии

- c голубой
- m фиолетовый
- y жёлтый
- r красный
- g зелёный
- b синий
- w белый
- k чёрный

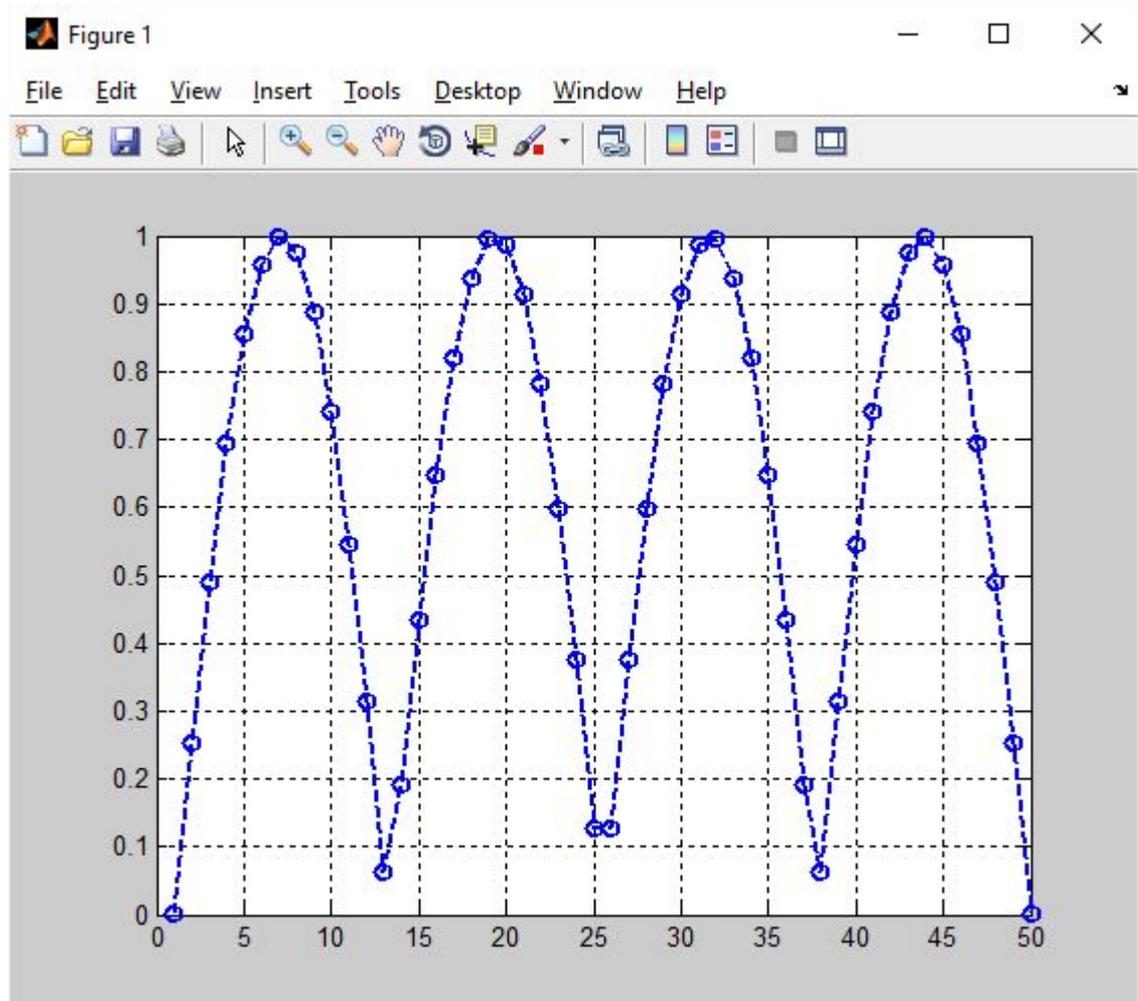
Построение графиков

```
clc, clear, close all  
x = linspace(-2*pi,2*pi,500);  
y = abs(sin(x));  
plot(y, 'LineWidth', 2)  
grid on
```



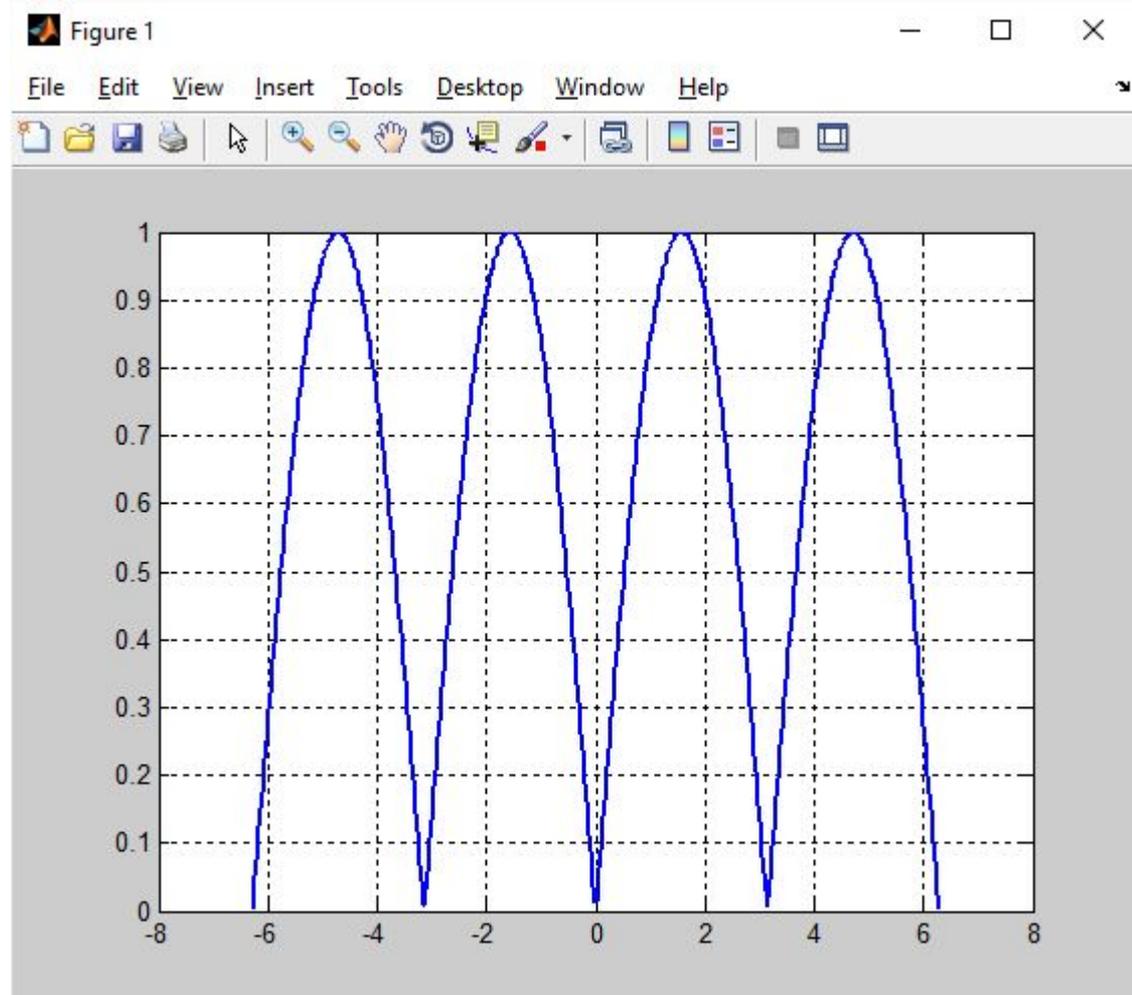
Построение графиков

```
clc, clear, close all
x = linspace(-2*pi,2*pi,50);
y = abs(sin(x));
plot(y, '--o', 'LineWidth', 2)
grid on
```



Построение графиков

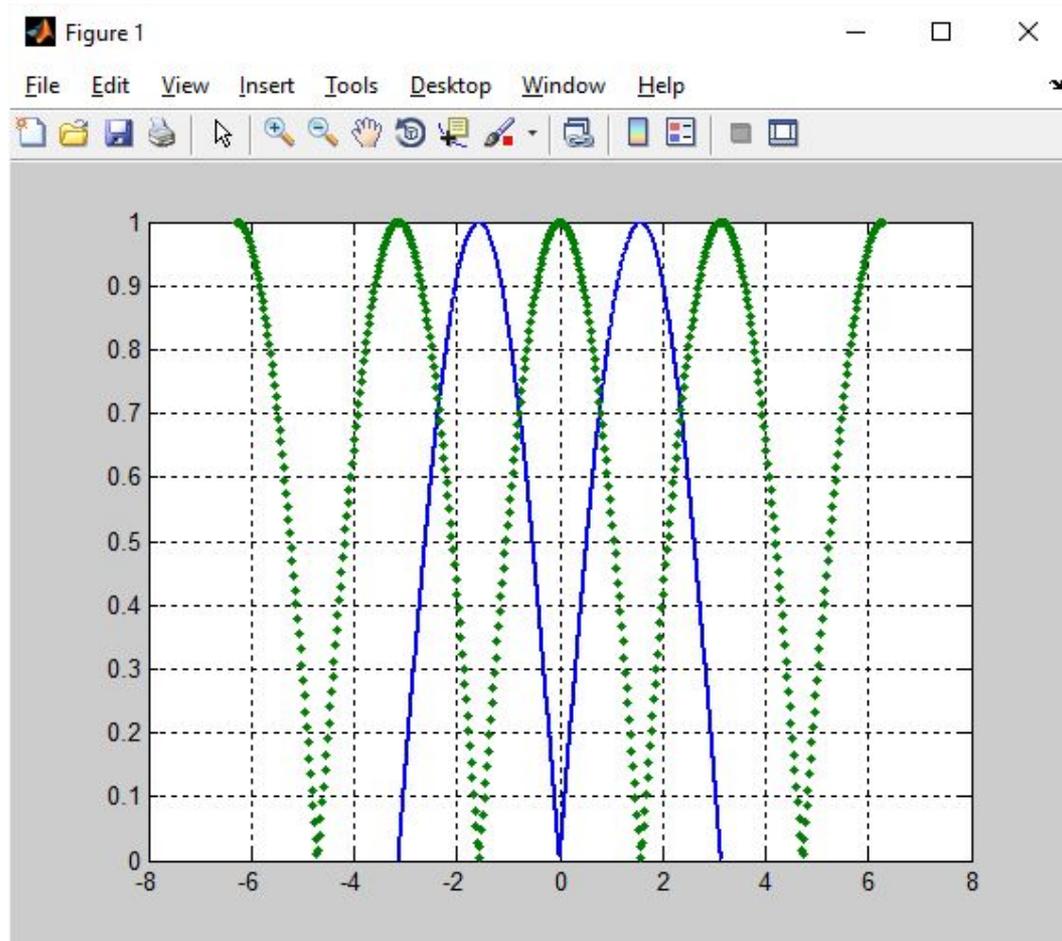
```
clc, clear, close all  
x = linspace(-2*pi,2*pi,500);  
y = abs(sin(x));  
plot(x,y,'LineWidth',2)  
grid on
```



Построение графиков

```
clc, clear, close all
x1 = linspace(-pi,pi,500);
x2 = linspace(-2*pi,2*pi,500);
y1 = abs(sin(x1)); y2 = abs(cos(x2));
plot(x1,y1,x2,y2, '.', 'LineWidth',2)
grid on
```

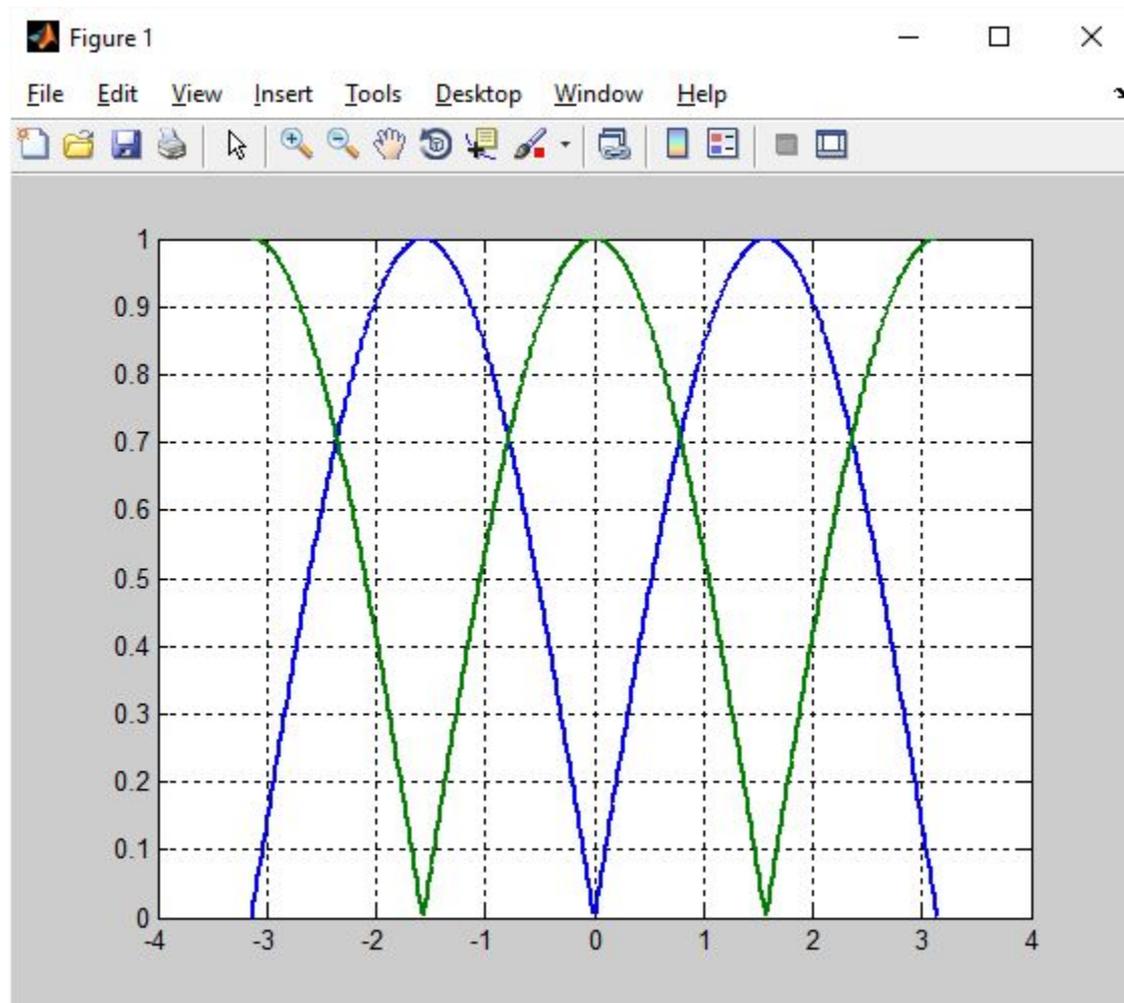
Ось X
индивидуальная



Построение графиков

```
clc, clear, close all
x = linspace(-pi,pi,500);
y1 = abs(sin(x));
y2 = abs(cos(x));
y = [y1;y2];
plot(x,y,'LineWidth',2)
grid on
```

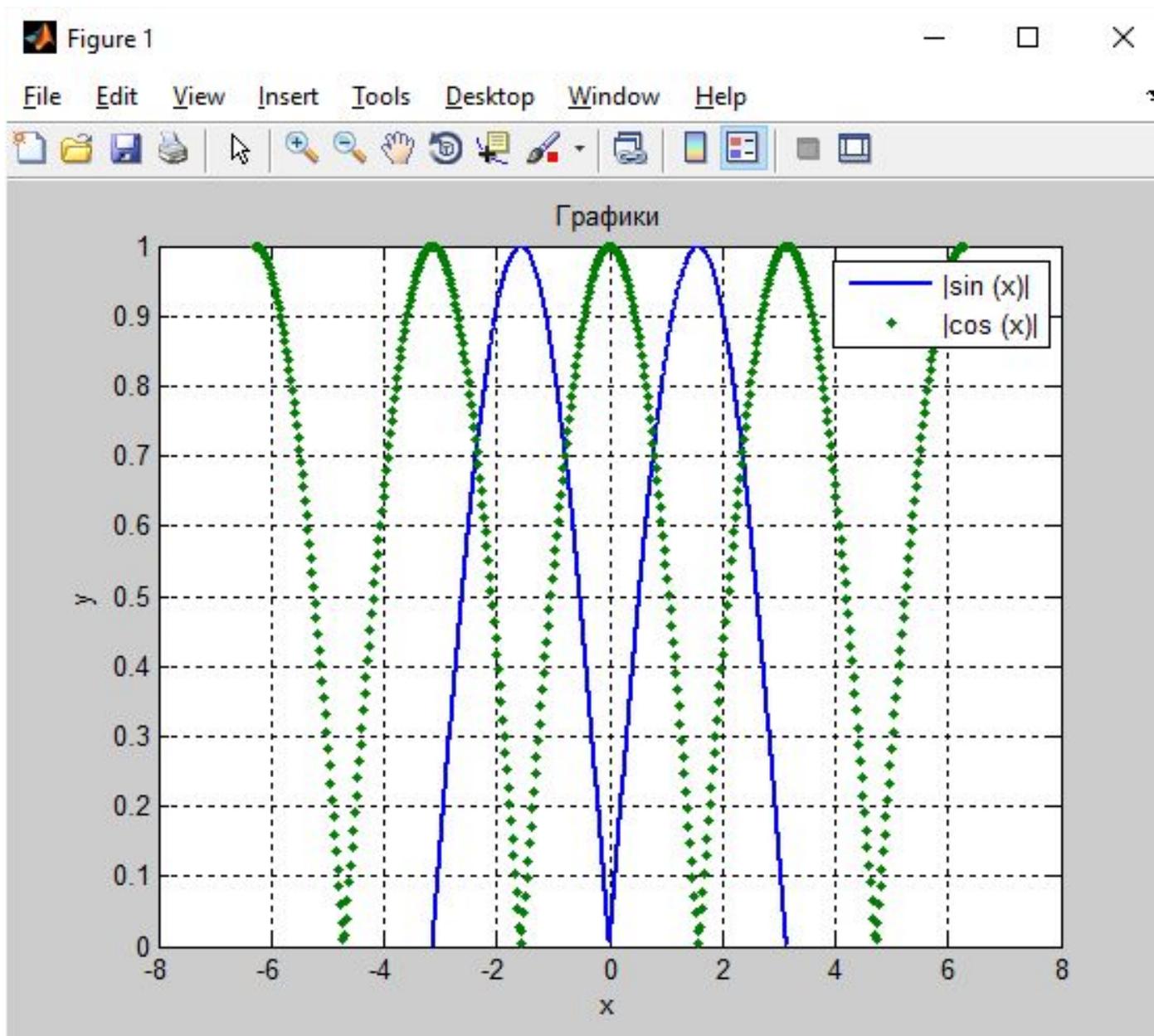
Ось x общая



Построение графиков

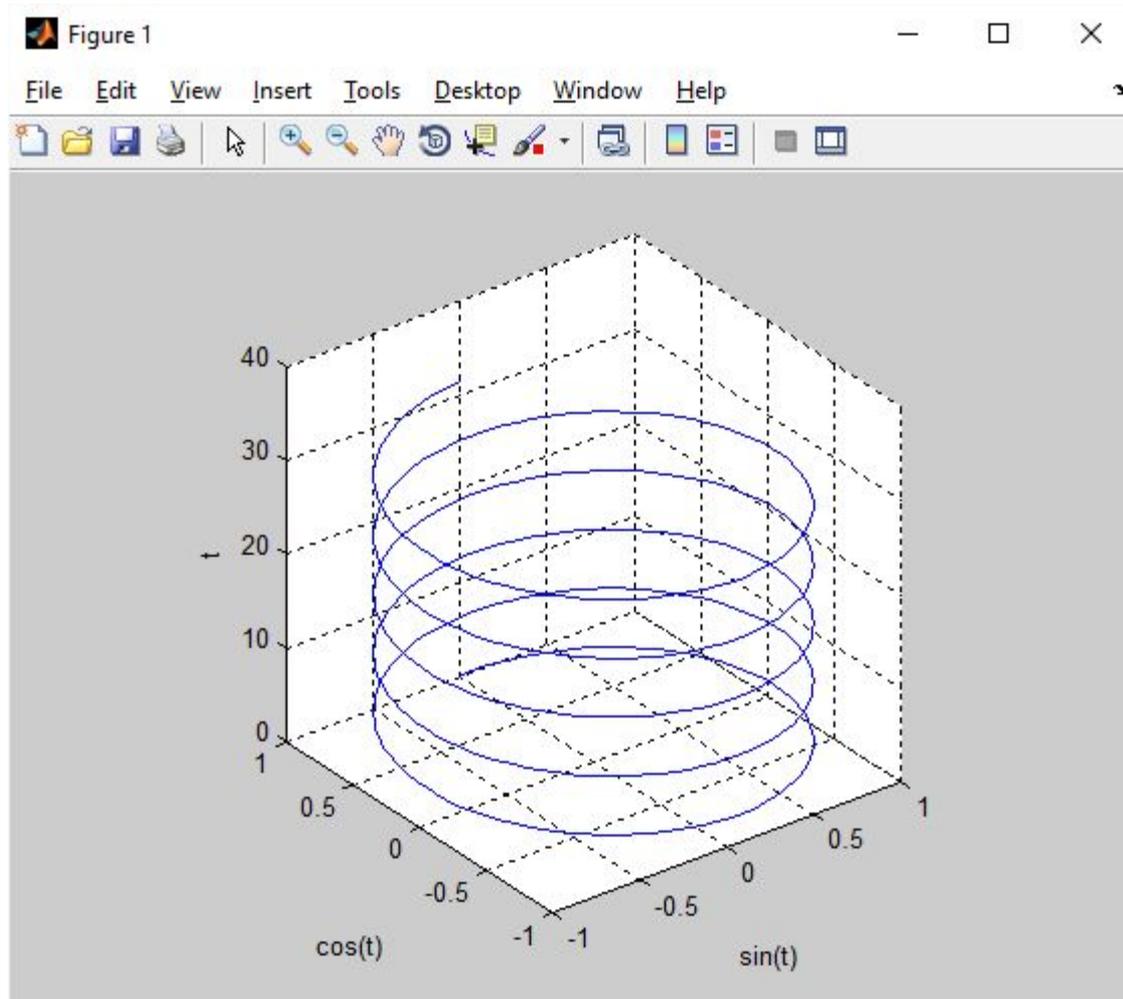
```
clc, clear, close all
x1 = linspace(-pi,pi,500);
x2 = linspace(-2*pi,2*pi,500);
y1 = abs(sin(x1));
y2 = abs(cos(x2));
plot(x1,y1,x2,y2, '.', 'LineWidth', 2)
legend('|sin (x)|', '|cos (x)|')
xlabel('x')
ylabel('y')
title('Графики')
grid on
```

Построение графиков



Построение графиков

```
clc, clear, close all
t = linspace(0,10*pi,500);
plot3(sin(t),cos(t),t)
grid on, axis square
xlabel('sin(t)'), ylabel('cos(t)'), zlabel('t')
```



Построение графиков

Функция $[X, Y] = \text{meshgrid}(x, y)$ задает сетку на плоскости x - y в виде двумерных массивов X, Y , которые определяются одномерными массивами x и y .

```
>> x=0:0.2:1;
```

```
>> y=-1:0.2:0;
```

```
>> [X, Y] = meshgrid(x,y)
```

$X =$

0	0.2	0.4	0.6	0.8	1
0	0.2	0.4	0.6	0.8	1
0	0.2	0.4	0.6	0.8	1
0	0.2	0.4	0.6	0.8	1
0	0.2	0.4	0.6	0.8	1
0	0.2	0.4	0.6	0.8	1

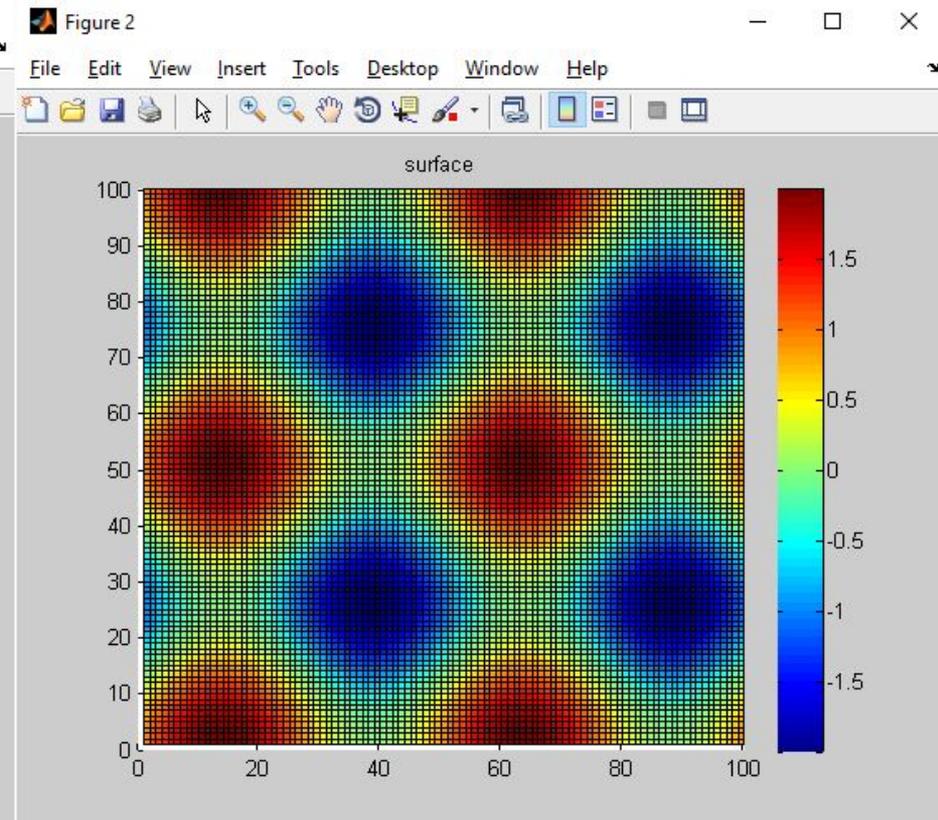
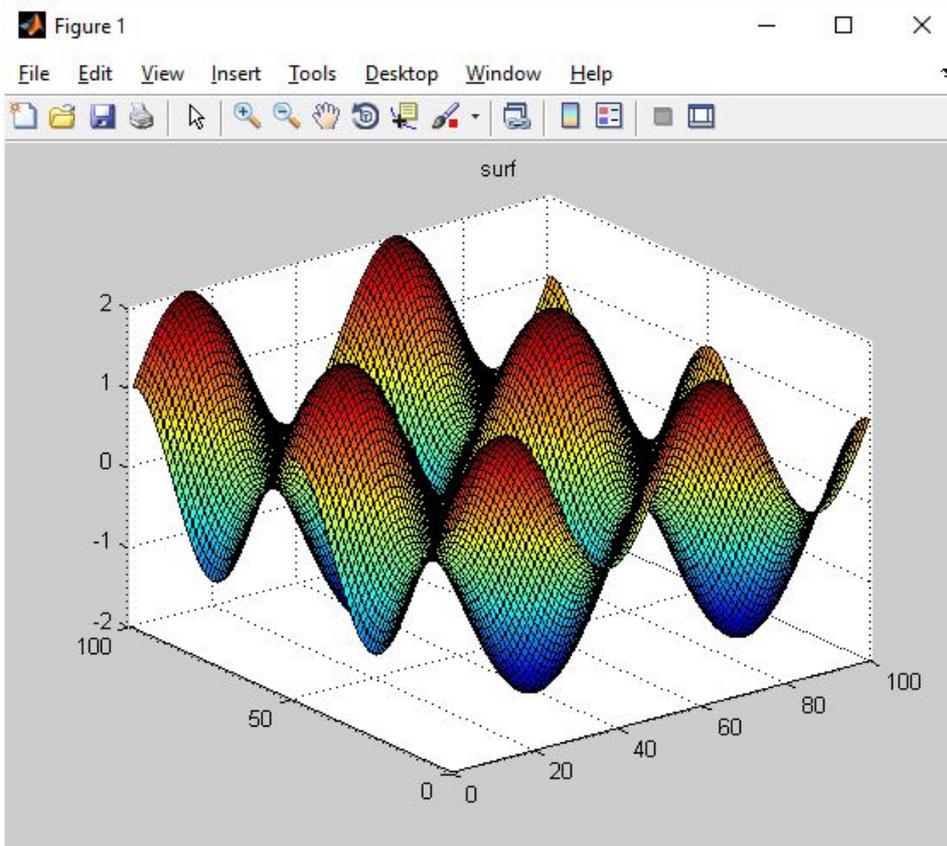
$Y =$

-1	-1	-1	-1	-1	-1
-0.8	-0.8	-0.8	-0.8	-0.8	-0.8
-0.6	-0.6	-0.6	-0.6	-0.6	-0.6
-0.4	-0.4	-0.4	-0.4	-0.4	-0.4
-0.2	-0.2	-0.2	-0.2	-0.2	-0.2
0	0	0	0	0	0

Построение графиков

```
clc, clear, close all
x = linspace(-2*pi,2*pi);
y = linspace(0,4*pi);
[X,Y] = meshgrid(x,y);
Z = sin(X) + cos(Y);
surf(Z)
title('surf')
figure
surface(Z)
title('surface')
```

Построение графиков

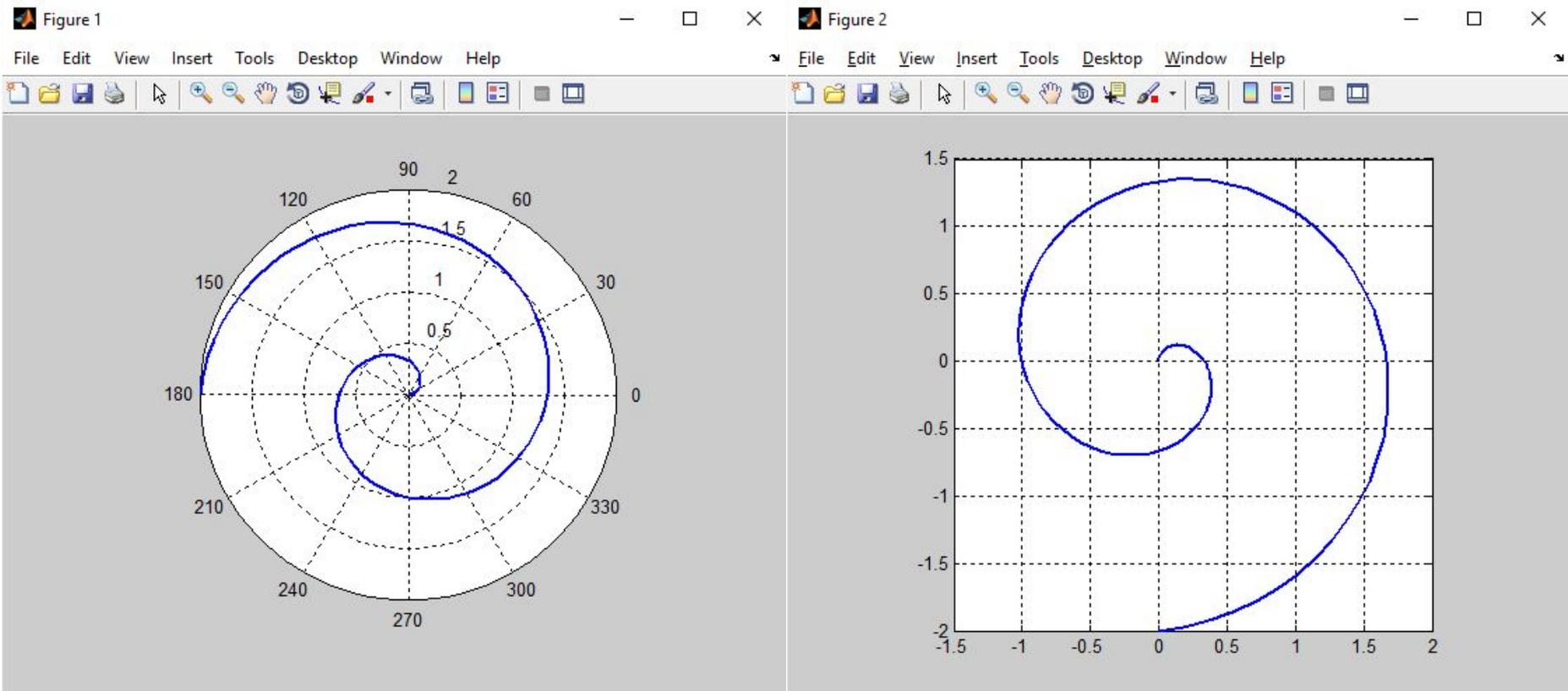


Построение графиков

```
clc, clear, close all  
phi = linspace(0, 3*pi, 100);  
ro = linspace(0, 2, 100);  
polar(phi, ro)
```

```
x = ro.*sin(phi);  
y = ro.*cos(phi);  
figure  
plot(x, y, 'LineWidth', 2)  
axis square  
grid on
```

Построение графиков



Построение графиков

```
%%сфера
```

```
phi = linspace(0, 2*pi, 50);
```

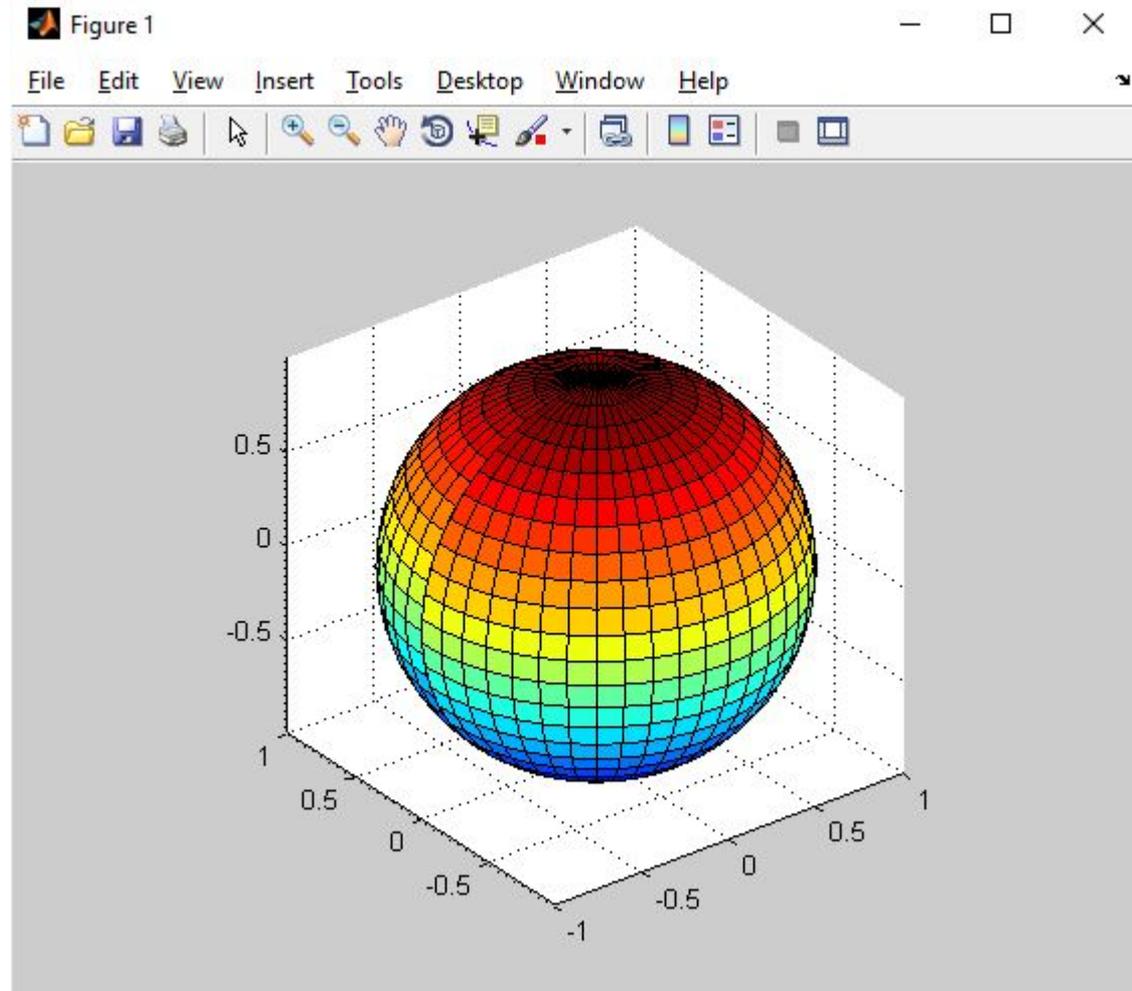
```
theta = linspace(0, pi, 25);
```

```
[Phi, Theta] = meshgrid( phi, theta );
```

```
[X_s,Y_s,Z_s] = sph2cart(Theta,Phi,1);
```

```
surf(X_s,Y_s,Z_s)
```

```
axis equal
```



ОПЕРАТОРЫ ОТНОШЕНИЯ

- < Меньше
- <= Меньше или равно
- > Больше
- > = Больше или равно
- == Равно тождественно
- ~ = Не равно

A = [2 7 6; 9 0 -1; 3 0.5 6];

B = [8 0.2 0; -3 2 5; 4 -1 7];

A < B

ans =

1	0	0
0	1	1
1	0	1

ОПЕРАТОРЫ ОТНОШЕНИЯ

A = 1;

B = [8 0.2 0; -3 2 5; 4 -1 7];

A < B

ans =

1 0 0

0 1 1

1 0 1

ЛОГИЧЕСКИЕ ОПЕРАТОРЫ

& И
| ИЛИ
~ НЕТ

```
u = [1 0 2 3 0 5];
```

```
v = [5 6 1 0 0 7];
```

```
u & v
```

```
ans =
```

```
1 0 1 0 0 1
```

```
and(u,v)
```

```
ans =
```

```
1 0 1 0 0 1
```

УПРАВЛЕНИЕ ПОСЛЕДОВАТЕЛЬНОСТЬЮ ИСПОЛНЕНИЯ ОПЕРАТОРОВ

Четыре основных оператора :

- оператор условия `if` (в сочетании с `else` и `elseif`);
- оператор переключения `switch` (в сочетании с `case` и `otherwise`);
- оператор условия `while`;
- оператор цикла `for`.

ОПЕРАТОР УСЛОВИЯ

```
if  
<логическое_выражение>  
    <инструкции>  
end
```

```
if <логическое_выражение>  
    <инструкции>  
else  
    <инструкции>  
end
```

```
if  
<логическое_выражение>  
    < инструкции>  
elseif <  
логическое_выражение>  
    <инструкции>  
else  
    <инструкции>  
end
```

```
if <логическое_выражение>  
    <инструкции>  
elseif <логическое_выражение>  
    <инструкции>  
elseif <логическое_выражение>  
    <инструкции>  
else  
    <инструкции>  
end
```

ОПЕРАТОР УСЛОВИЯ

if expression, statements, end evaluates an expression, and executes a group of statements when the expression is true. ***An expression is true when its result is nonempty and contains only nonzero elements (logical or real numeric). Otherwise, the expression is false.***

Выражение истинно, если его результат непуст и содержит только ненулевые элементы (логические или вещественные числовые). В противном случае выражение является ложным.

ОПЕРАТОР УСЛОВИЯ

Нужен ли условный оператор при решении квадратного уравнения в MATLAB?

```
a = 2; b = 1; c = 3;  
D=b^2-4*a*c;  
if D>0  
    x(1) = (-b+sqrt(D))/2/a;  
    x(2) = (-b-sqrt(D))/2/a;  
elseif D==0  
    x = -b/2/a,  
Else  
    x(1) = (-b+i*sqrt(abs(D)))/2/a;  
    x(2) = (-b-i*sqrt(abs(D)))/2/a;  
End  
disp('Корни')  
disp(x)
```

НЕ НУЖЕН

ОПЕРАТОР УСЛОВИЯ

Нужен ли условный оператор при решении квадратного уравнения в MATLAB?

$$a = 2; b = 1; c = 3;$$

$$D=b^2-4*a*c;$$

$$x(1) = (-b+sqrt(D))/2/a;$$

$$x(2) = (-b-sqrt(D))/2/a;$$

$$x =$$

$$-0.2500 + 1.1990i \quad -0.2500 - 1.1990i$$

$$a = 1; b = 2; c = 1;$$

$$D=b^2-4*a*c;$$

$$x(1) = (-b+sqrt(D))/2/a;$$

$$x(2) = (-b-sqrt(D))/2/a;$$

$$x =$$

$$-1 \quad -1$$

ОПЕРАТОР УСЛОВИЯ

Решение уравнений и систем уравнений

```
clc
```

```
clear all
```

```
syms x
```

```
Res = solve(x^2 + 2*x + 2)
```

```
Res =
```

```
- 1 - i
```

```
- 1 + i
```

ОПЕРАТОР УСЛОВИЯ

```
A = 1;  
if A == 1  
    disp('expression is true');  
else  
    disp('expression is false');  
end
```

expression is true

```
A = [1 2 3 4];  
if A == 1  
    disp('expression is true');  
else  
    disp('expression is false');  
end
```

expression is false

ОПЕРАТОР УСЛОВИЯ

```
A = [1 2 3 4];
```

```
B = [1 2 3 5];
```

```
if A == B
```

```
    disp('expression is true');
```

```
else
```

```
    disp('expression is false');
```

```
end
```

expression is false

ОПЕРАТОР ПЕРЕКЛЮЧЕНИЯ

Синтаксис:

```
switch <выражение>
case <значение1>
    инструкции
case <значение2>
    инструкции
...
Otherwise
    инструкции
end
```

```
key = input('Введите 1,2,3? ', 's');
switch key
case '1'
    disp('Вы выбрали 1');
case '2'
    disp('Вы выбрали 2');
case '3'
    disp('Вы выбрали 3');
otherwise
    disp('Вы - странный');
end
```

ОПЕРАТОР ЦИКЛА С НЕОПРЕДЕЛЕННЫМ ЧИСЛОМ ОПЕРАЦИЙ

Синтаксис:

```
while <выражение>  
    <инструкции>  
end
```

```
a=5; i=2;  
    while i<10  
        a = a - 2;  
        i = i + 3;  
    end  
disp(a)
```

ОПЕРАТОР ЦИКЛА С ОПРЕДЕЛЕННЫМ ЧИСЛОМ ОПЕРАЦИЙ

Синтаксис:

```
for <перем. цикла> = <нач.  
знач>:<приращение>:<конеч.знач.>  
    <инструкции>  
end
```

```
x(1) = 1;  
for i = 2:6  
    x(i) = 2*x(i-1);  
end
```

```
for i = 1:m  
    for j = 1:n  
        A(i,j) = 1/(i + j - 1);  
    end  
end
```

СКОРОСТЬ ВЫЧИСЛЕНИЯ МАТЛАВ

Рассчитать значение функции

$$y(x) = \sin 3x, \quad x \in [0; 100\pi]$$

```
dx = pi/30;  
nx = 1 + 100*pi/dx;  
for i = 1:nx  
    x(i) = (i-1)*dx;  
    y(i) = sin(3*x(i));  
end
```

0.166 сек.

```
dx = pi/30;  
nx = 1 + 100*pi/dx;  
x=zeros(1,nx);  
y=zeros(1,nx);  
for i = 1:nx  
    x(i) = (i-1)*dx;  
    y(i) = sin(3*x(i));  
end
```

0.018 сек. (в 9 раз)

```
x = 0:pi/30:100 * pi;  
y = sin(3*x);
```

0.0045 сек. (в 37 раз)

Типы m-файлов

M-сценарии

Не использует входных и выходных аргументов

Оперирует с данными из рабочей области

Предназначен для автоматизации последовательности шагов, которые нужно выполнять много раз

M-функции

Использует входные и выходные аргументы

По умолчанию, внутренние переменные являются локальными по отношению к функции

Предназначена для расширения возможностей языка MATLAB (библиотеки функций, пакеты прикладных программ)

Типы m-файлов

Требования к названию m-файлов:

- имя файла начинается с латинской буквы, далее буквы, цифры подчеркивания
- имя файла не должно превышать 31 символ
- различаются символы верхнего и нижнего регистра

Типы m-файлов. Структура m-функции

Структура m-функции:

- строки определения функции;
- первой строки комментария;
- собственно комментария;
- тела функции;
- строчных комментариев.

Типы m-файлов. Пример

Разработать m-сценарий и m-функцию для вычисления среднего из двух чисел

M-сценарий

A=10;

B=12;

C = (A+B)/2;

Типы m-файлов. Пример M-функция

Содержимое файла *my_avg.m*

```
function [ C ] = my_avg( x,y)
```

```
% my_avg ср. знач. двух чисел
```

```
[mx, nx] = size(x);
```

```
[my, ny] = size(y);
```

```
if (mx == my & nx == ny |
```

```
    (mx == 1 & nx == 1) | (my == 1 & ny == 1))
```

```
    C = (x+y)/2;
```

```
else
```

```
    error('X и Y разной размерности');
```

```
end
```

```
end
```

M-сценарий или командная строка

```
A=10;
```

```
B=12;
```

```
C = my_avg(A,B);
```

Определение
функции

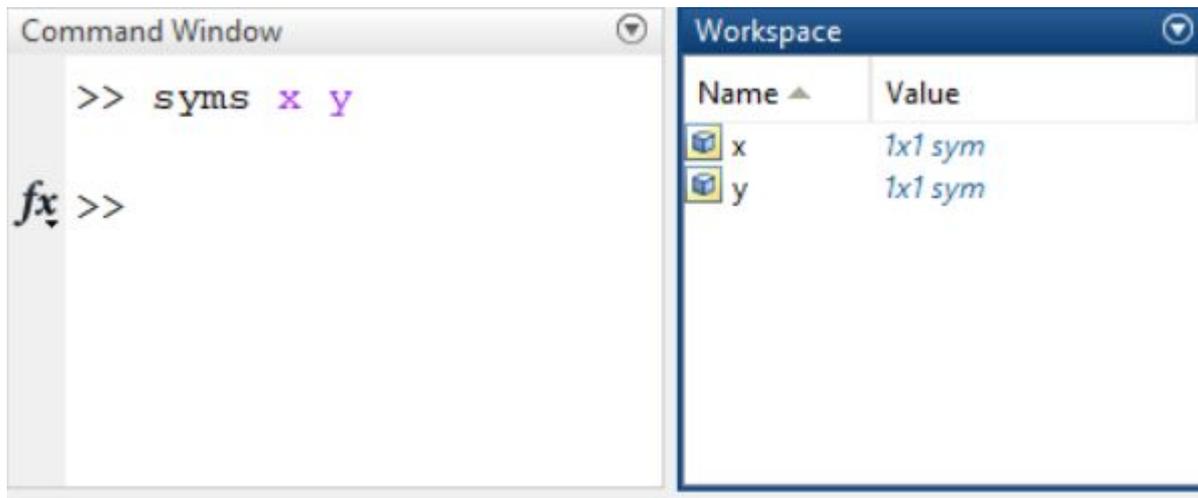
Комментарий к
функции для команд
doc и help

Тело функции

Symbolic MathToolBox

Объявление символьных переменных

`syms x y`



Symbolic MathToolBox

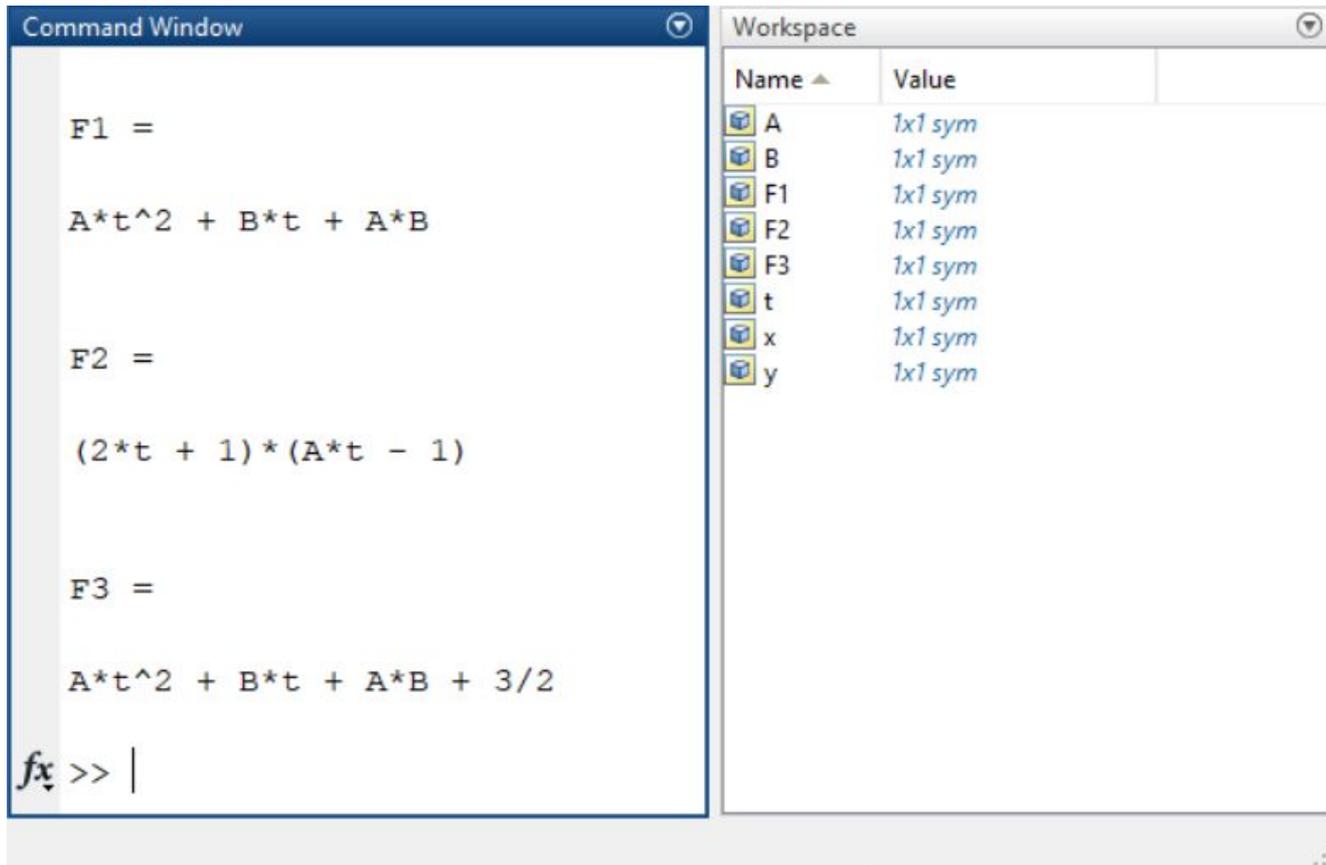
Символьные выражения

```
syms A B t
```

```
F1 = A*t^2 + B*t + A*B
```

```
F2 = (2*t+1)*(A*t-1)
```

```
F3 = F1+1.5
```



The screenshot shows the MATLAB Command Window and Workspace. The Command Window displays the following code and results:

```
F1 =  
  
A*t^2 + B*t + A*B  
  
F2 =  
  
(2*t + 1)*(A*t - 1)  
  
F3 =  
  
A*t^2 + B*t + A*B + 3/2  
  
fx >> |
```

The Workspace window shows the following variables and their values:

Name	Value
A	1x1 sym
B	1x1 sym
F1	1x1 sym
F2	1x1 sym
F3	1x1 sym
t	1x1 sym
x	1x1 sym
y	1x1 sym

Symbolic MathToolBox

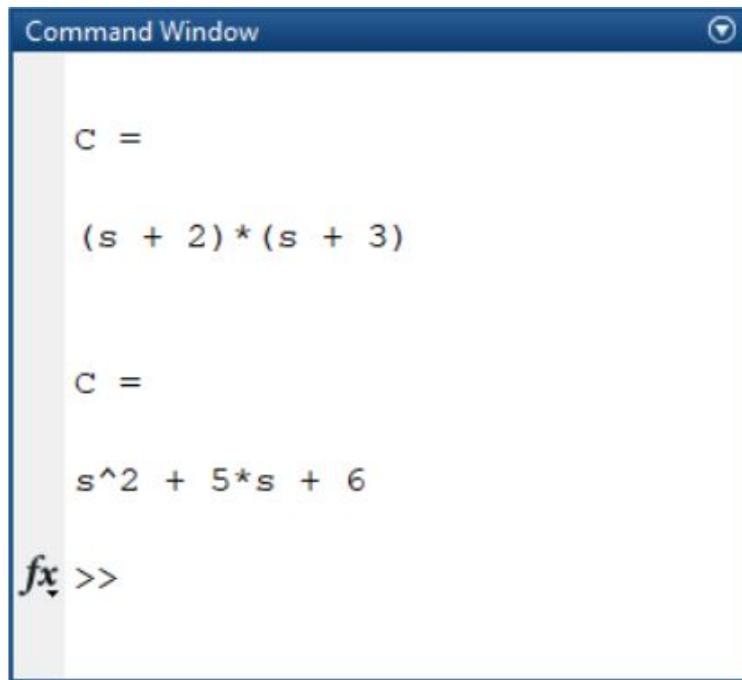
Алгебраические манипуляции

<code>expand(S)</code>	Раскрывает скобки в выражении S
<code>factor(S)</code>	Разлагает на множители выражение S
<code>simplify(S)</code>	Упрощает каждый элемент символьной матрицы S
<code>subs(S, oldvar, newvar)</code>	Заменяет в выражении S каждое вхождение символической переменной <code>oldvar</code> новой переменной <code>newvar</code>

Symbolic MathToolBox

Раскрытие скобок

```
syms s  
A = s + 2;  
B = s + 3;  
C = A*B  
C = expand(C)
```



```
Command Window  
  
C =  
  
(s + 2) * (s + 3)  
  
C =  
  
s^2 + 5*s + 6  
  
fx >>
```

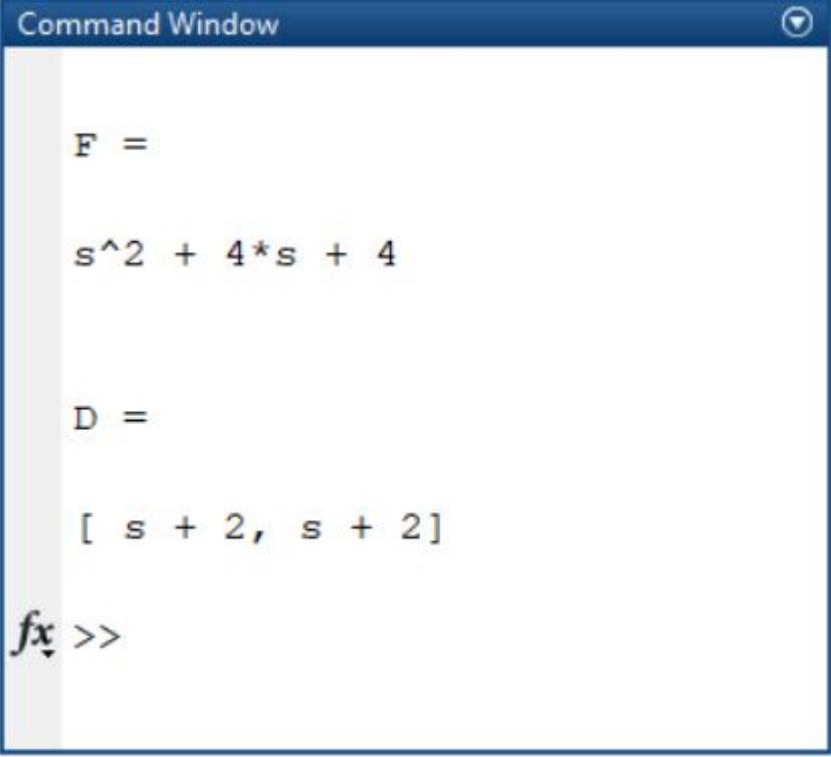
Symbolic MathToolBox

Разложение на множители

```
syms s
```

```
F = s^2 + 4*s + 4
```

```
D = factor(F)
```



```
Command Window
```

```
F =  
  
s^2 + 4*s + 4  
  
D =  
  
[ s + 2, s + 2]  
  
fx >>
```

Symbolic MathToolBox

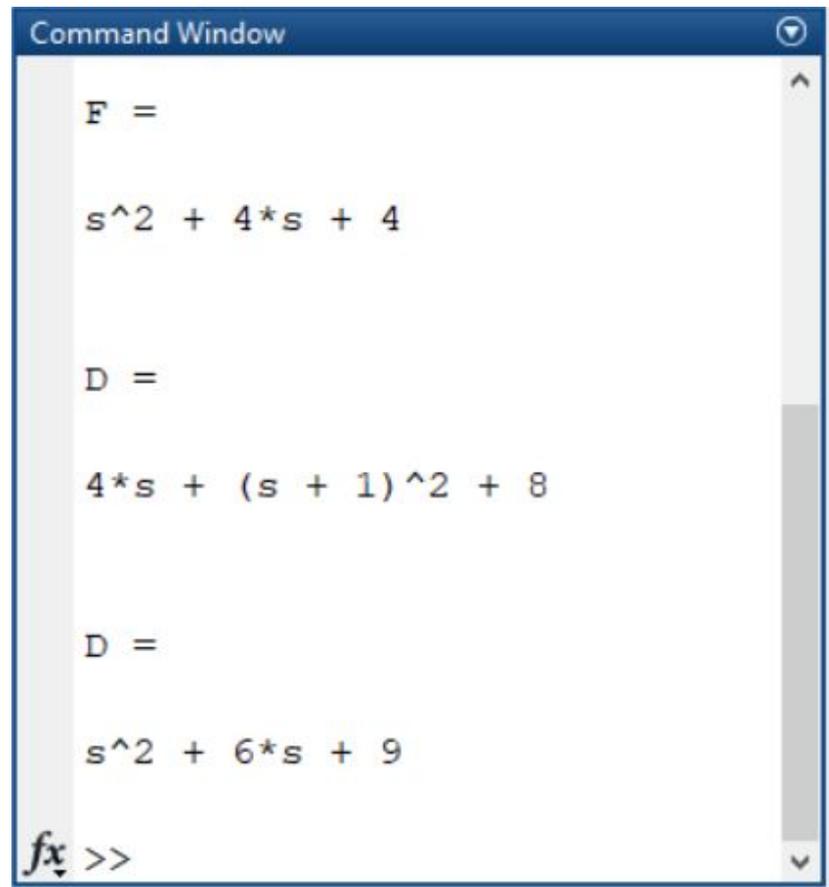
Подстановка переменной

```
syms s
```

```
F = s^2 + 4*s + 4
```

```
D = subs(F, s, s+1)
```

```
D = expand(D)
```



```
Command Window
```

```
F =  
  
s^2 + 4*s + 4  
  
D =  
  
4*s + (s + 1)^2 + 8  
  
D =  
  
s^2 + 6*s + 9  
  
fx >>
```

Symbolic MathToolBox

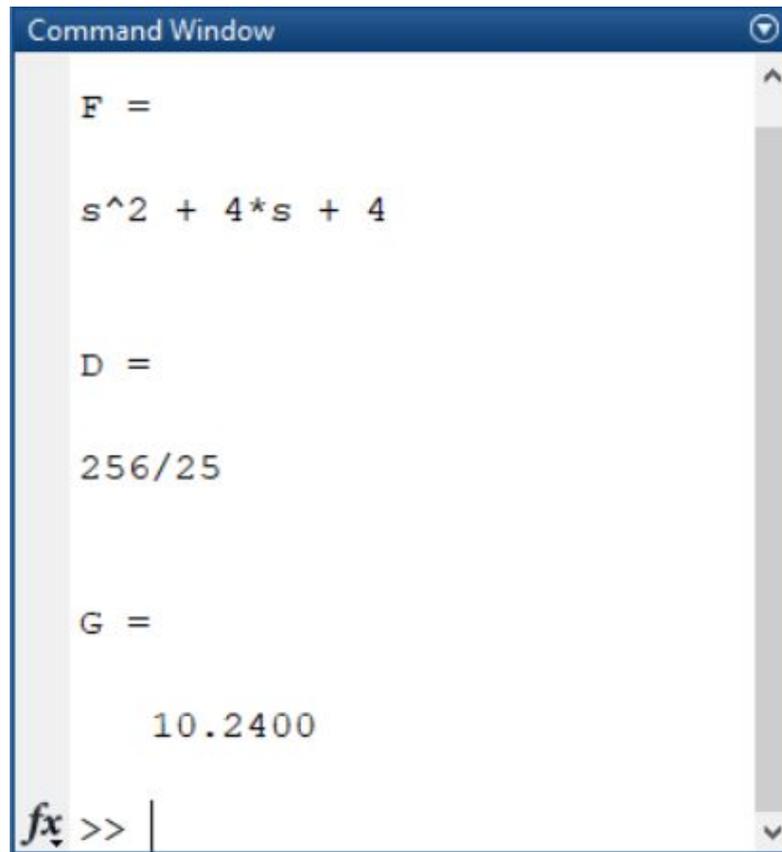
Вычисление символьных значений

```
syms s
```

```
F = s^2 + 4*s + 4
```

```
D = subs(F, s, 1.2)
```

```
G = double(D)
```



```
Command Window
```

```
F =  
  
s^2 + 4*s + 4  
  
D =  
  
256/25  
  
G =  
  
10.2400  
  
fx >> |
```

Symbolic MathToolBox

Построить частотные характеристики передаточной

функции

```
clc; clear all; close all
```

```
syms p w real K real T real
```

```
W = K/(T*p + 1) % ПФ W=K/(Tp+1)
```

```
W = subs(W, K, 2) % W=2/(Tp+1)
```

```
W = subs(W, T, 4) % W=2/(4p+1)
```

```
H = subs(W, p, i*w) % W=2/(4jw+1)
```

```
x = -2:0.1:1; fr = 10.^x;
```

```
H = subs(H, w, fr); % АФХ
```

```
L = 20*log10(double(abs(H))); % ЛАЧХ
```

```
F = rad2deg(double(angle(H))); % ФЧХ
```

```
subplot(2,1,1)
```

```
semilogx(fr, L) % логарифмический масштаб X
```

```
title('magnitude');
```

```
xlabel('lg\omega'); ylabel('L(\omega)');
```

```
grid on
```

```
subplot(2,1,2)
```

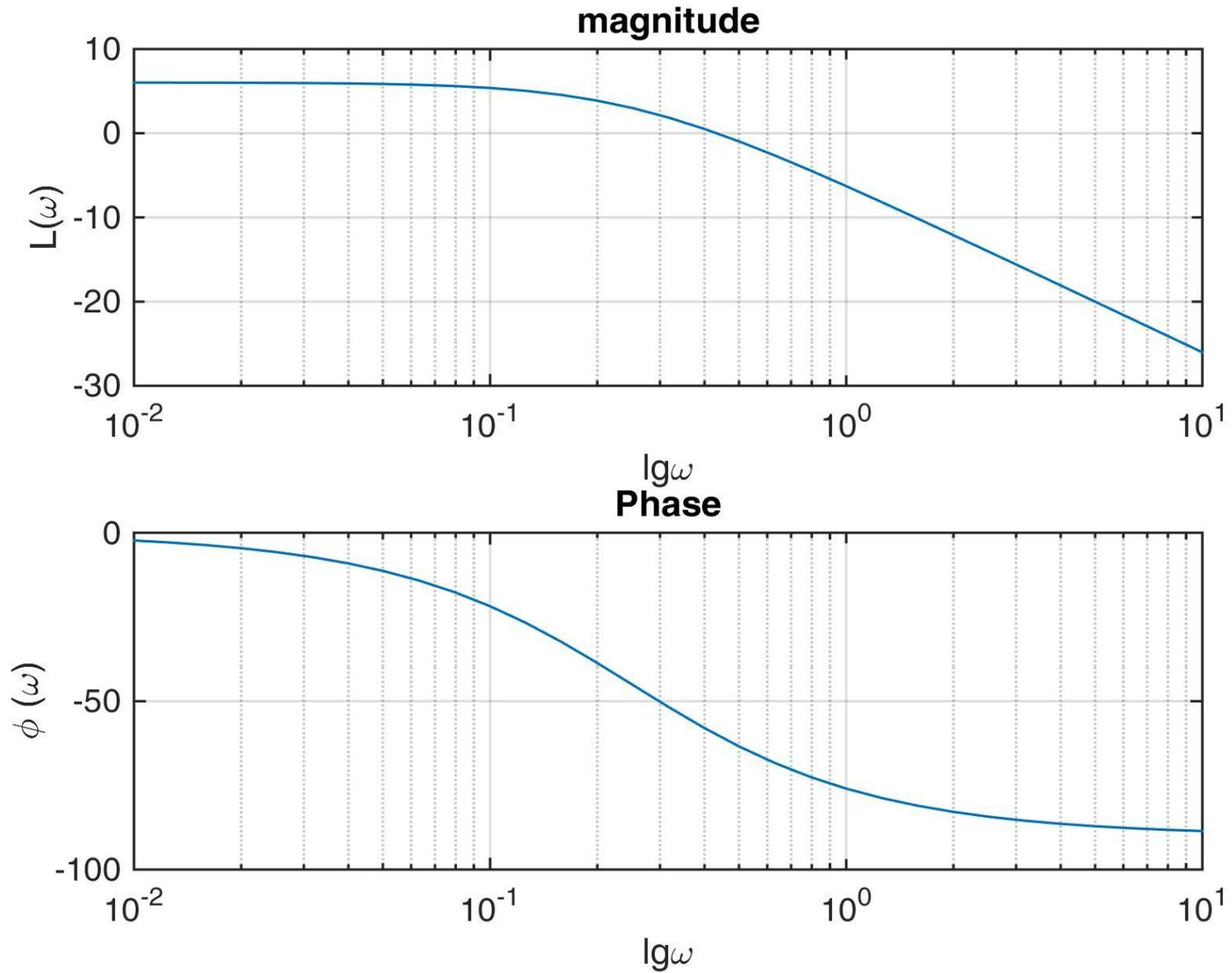
```
semilogx(fr, F)
```

```
title('Phase');
```

```
xlabel('lg\omega'); ylabel('\phi (\omega)');
```

```
grid on
```

Symbolic MathToolBox



Контрольная точка 1

Состав теста:

- 1) Операции над матрицами в MATLAB, 4 шт (средняя сложность);
- 2) цикл с определенным числом шагов, 2 шт (средняя сложность);
- 3) цикл с не определенным числом операций, 2 (средняя сложность).

Каждое задание – 12.5% рейтинга

Время выполнения теста – 30 минут.

Все задания открытые.