

Лекция №5:

Основы графической визуализации вычислений

Учебные вопросы:

1. Построение графика функций одной переменной
2. Столбцовые диаграммы.
3. Построение трехмерных графиков.

Рекомендуемая литература:

(L#5)



1. **Половко А.М., Бутусов П.Н. MATLAB для студентов. – СПб.: БХВ-Петербург, 2005. – 320 с.**
2. Дьяконов В.П. MATLAB 6: учебный курс. – СПб.: Питер, 2001. – 592 с.
3. Дьяконов В.П., Круглов В. Математические пакеты расширения MATLAB. Специальный справочник.
4. **Дьяконов В.П. MATLAB 6.5 SP1/7.0 + Simulink 5/6. Основы применения. Серия «Библиотека профессионала». – М.: СОЛОН-Пресс, 2005. – 800 с.**
5. Дьяконов В.П. MATLAB 6.5 SP1/7 + Simulink 5/6R в математике и моделировании. Серия Библиотека профессионала. – М.: СОЛОН-Пресс, 2005. – 576 с.
6. **Потемкин В. Г. Система инженерных и научных расчетов MATLAB 5. X: в 2-х т.**
7. Чен К., Джиблин П. Ирвинг А. MATLAB в математических исследованиях: Пер. с англ. – М.: Мир, 2001. – 346 с.
8. **Ануфриев И.Е., Смирнов А.Б., Смирнова Е.Н. MATLAB 7. – СПб.: БХВ-Петербург, 2005. – 1104 с.**



Введение

Начиная с версии MATLAB 4.0, впервые ориентированной на Windows, графические средства системы MATLAB были существенно улучшены. Основные отличительные черты графики в новой версии MATLAB 6:

- существенно улучшенный интерфейс графических окон;
- возможность создания графики в отдельных окнах и вывода нескольких графических окон;
- задание различных координатных систем и осей;
- высокое качество графики и широкие возможности использования цвета;
- легкость установки графических признаков — атрибутов;
- обилие параметров команд графики;
- возможность получения естественно выглядящих трехмерных фигур и их сочетаний;
- простота построения трехмерных графиков с их проекцией на плоскость;
- возможность построения сечений трехмерных фигур и поверхностей плоскостями;
- функциональная многоцветная и полутоновая окраска;
- возможность создания анимационной графики;
- возможность создания объектов для типового интерфейса пользователя.

1. Построение графика функций одной переменной

Синтаксис команды

```
plot(Y)
```

```
plot(X1,Y1,...)
```

```
plot(X1,Y1,LineSpec,...)
```

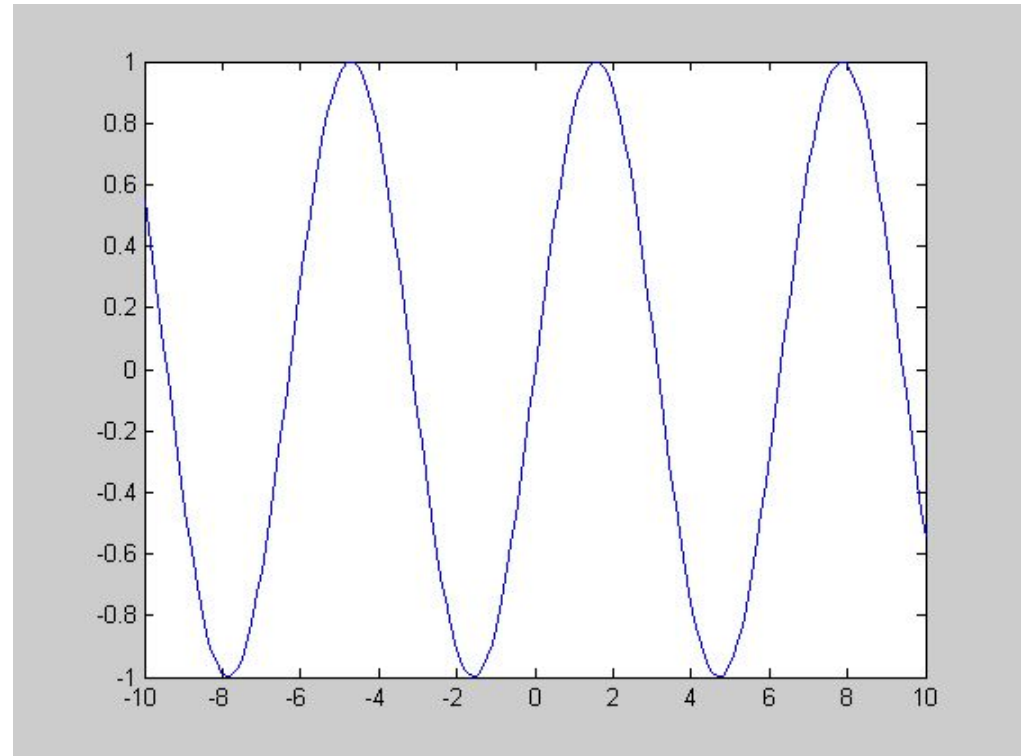
Пример

```
clear all
```

```
x=-10:0.1:10;
```

```
y=sin(x);
```

```
plot(x,y)
```



Примеры построения графика функций одной переменной

Пример 2

```
x=-10:0.1:10;
```

```
y=tan(x);
```

```
plot(x,y)
```

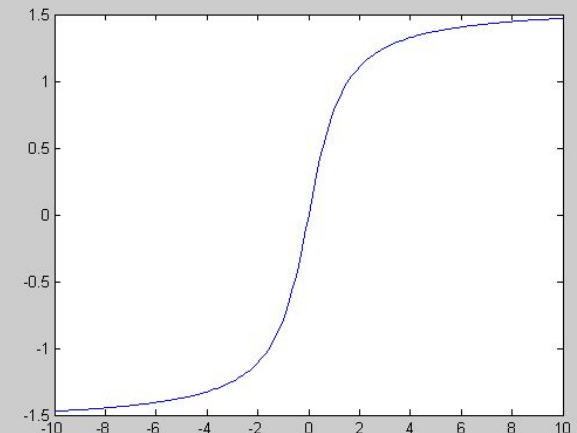
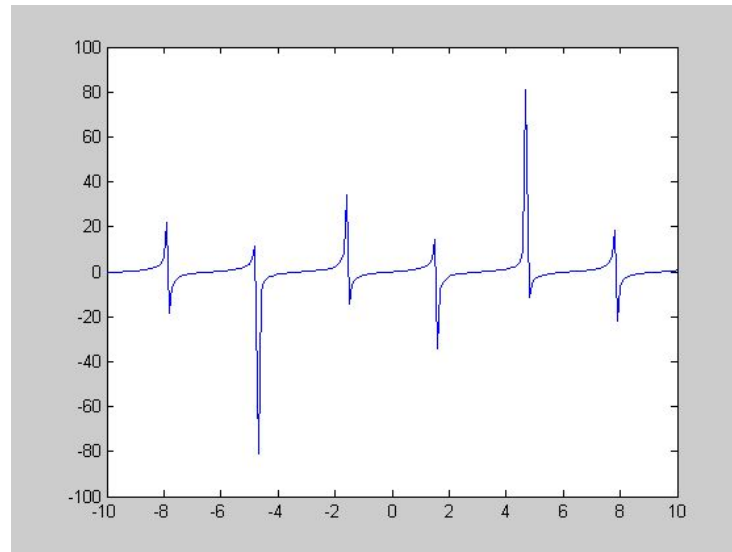
Пример 3

```
clear all
```

```
x=-10:0.1:10;
```

```
y=atan(x);
```

```
plot(x,y)
```

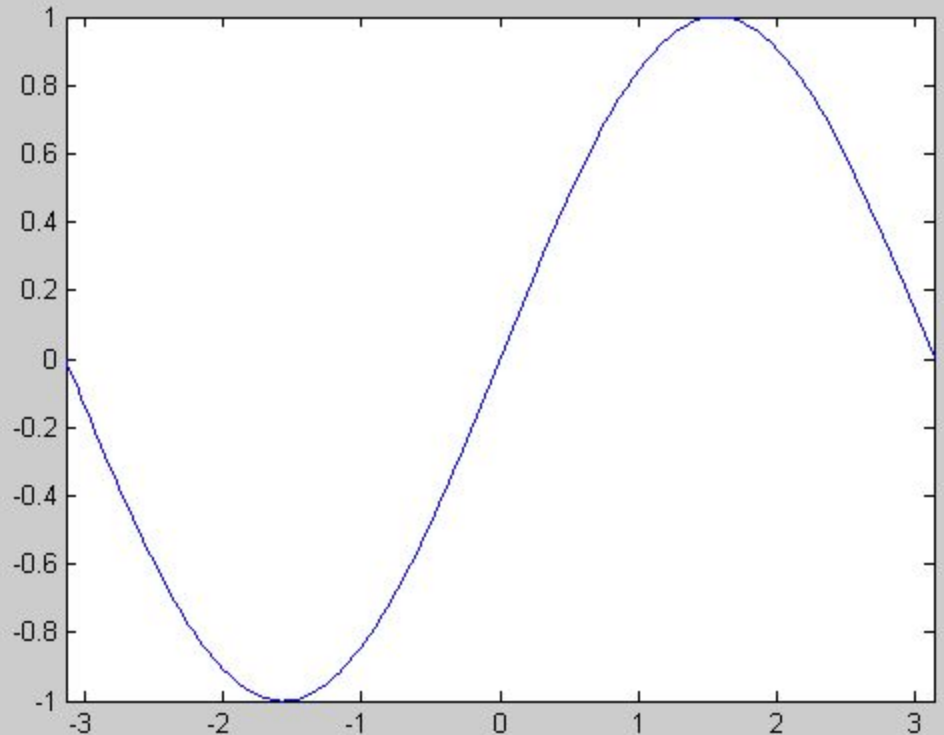


Функция fplot

Пример 4

```
clear all
```

```
fplot('sin',[-pi,pi])
```



Построение в одном окне графиков нескольких функций

Пример 5

```
clear all
```

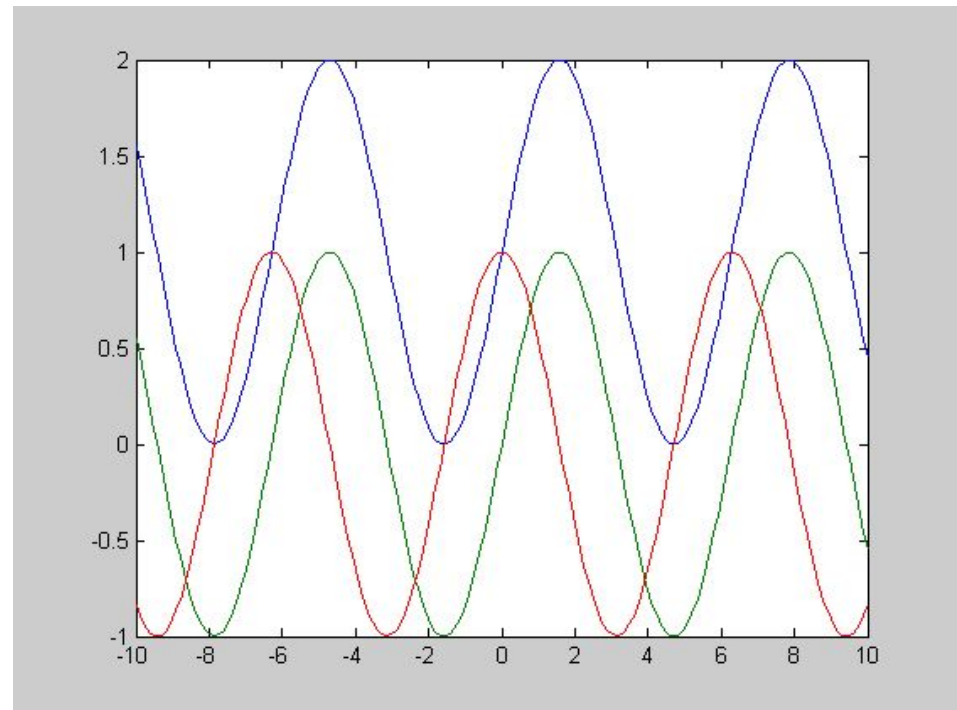
```
x=-10:0.1:10;
```

```
y1=sin(x)+1;
```

```
y2=sin(x);
```

```
y3=cos(x);
```

```
plot(x,y1,x,y2,x,y3)
```



Построение в одном окне графиков нескольких функций

Синтаксис команд

```
subplot(m,n,p)
```

Пример 6

```
clear all;
```

```
x=-10:0.1:10;
```

```
y1=sin(x)+1;
```

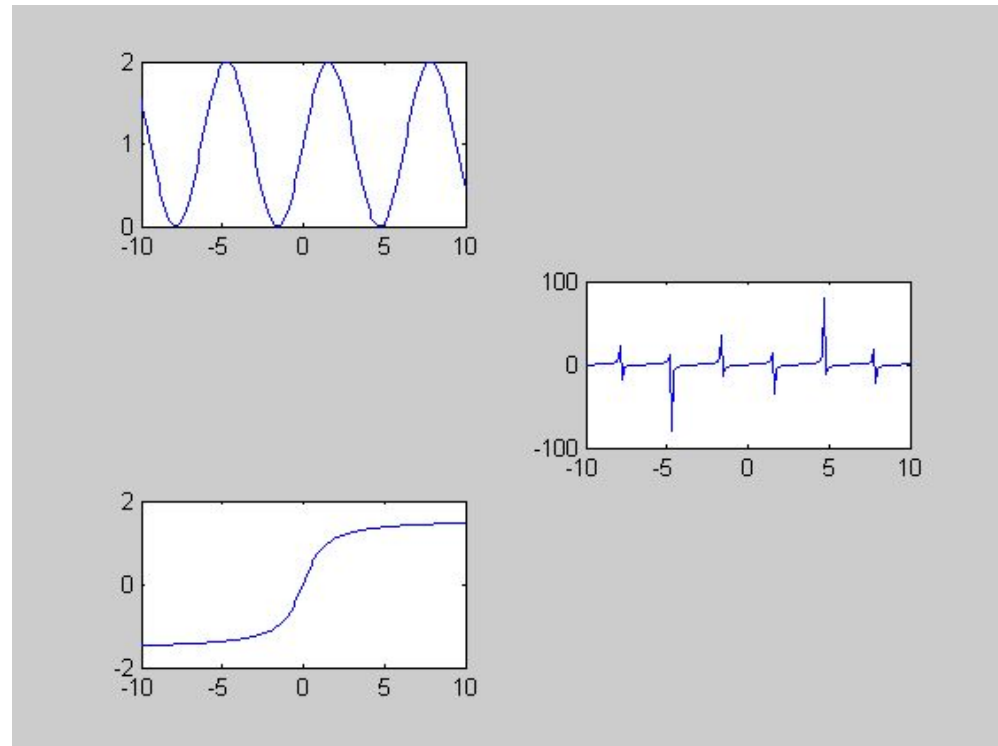
```
y2=tan(x);
```

```
y3=atan(x);
```

```
subplot(3,2,1); plot(x,y1);
```

```
subplot(3,2,4); plot(x,y2);
```

```
subplot(3,2,5); plot(x,y3);
```



Форматирование графиков

Названия осей xlabel, ylabel

Синтаксис команд

```
xlabel('string')
```

Пример 7

```
clear all
```

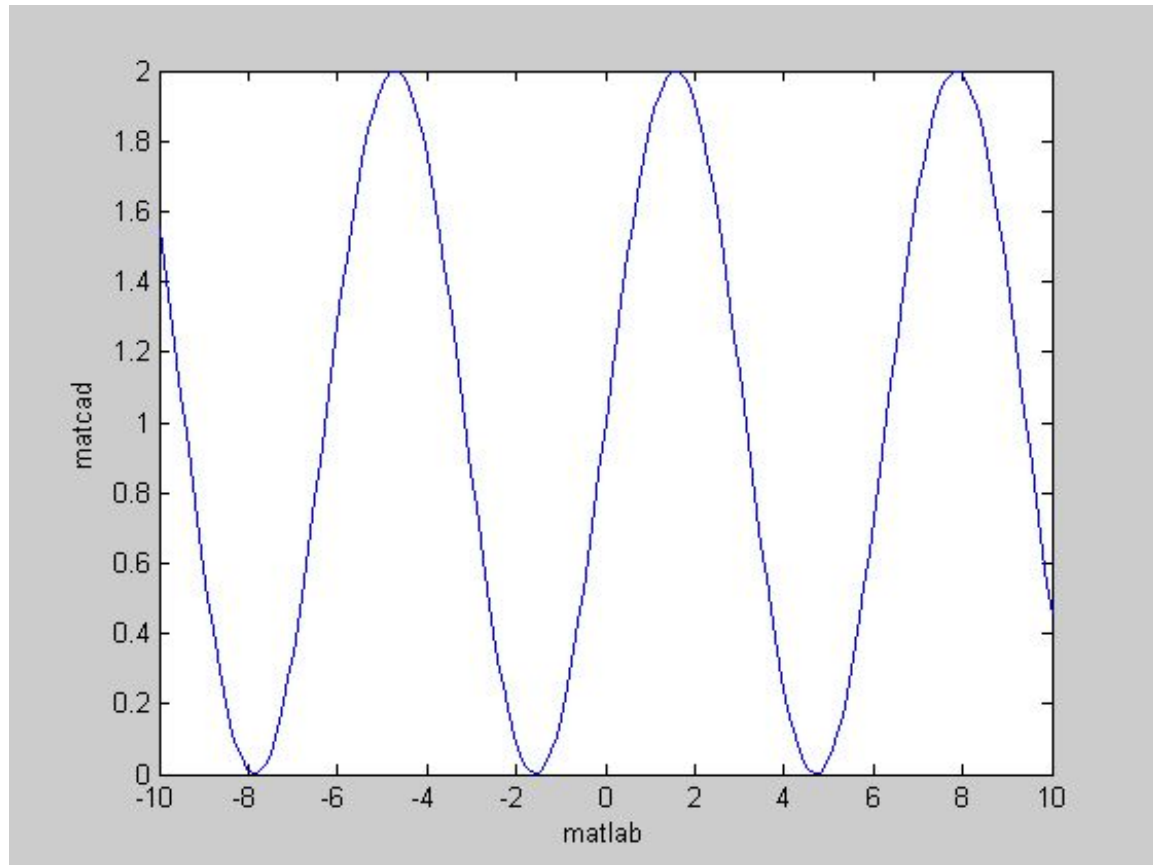
```
x=-10:0.1:10;
```

```
y=sin(x)+1;
```

```
plot(x,y)
```

```
xlabel('matlab')
```

```
ylabel('matcad')
```



Форматирование графиков

Название рисунка

Синтаксис команды

```
title('string')
```

Пример 8

```
clear all
```

```
x=-10:0.1:10;
```

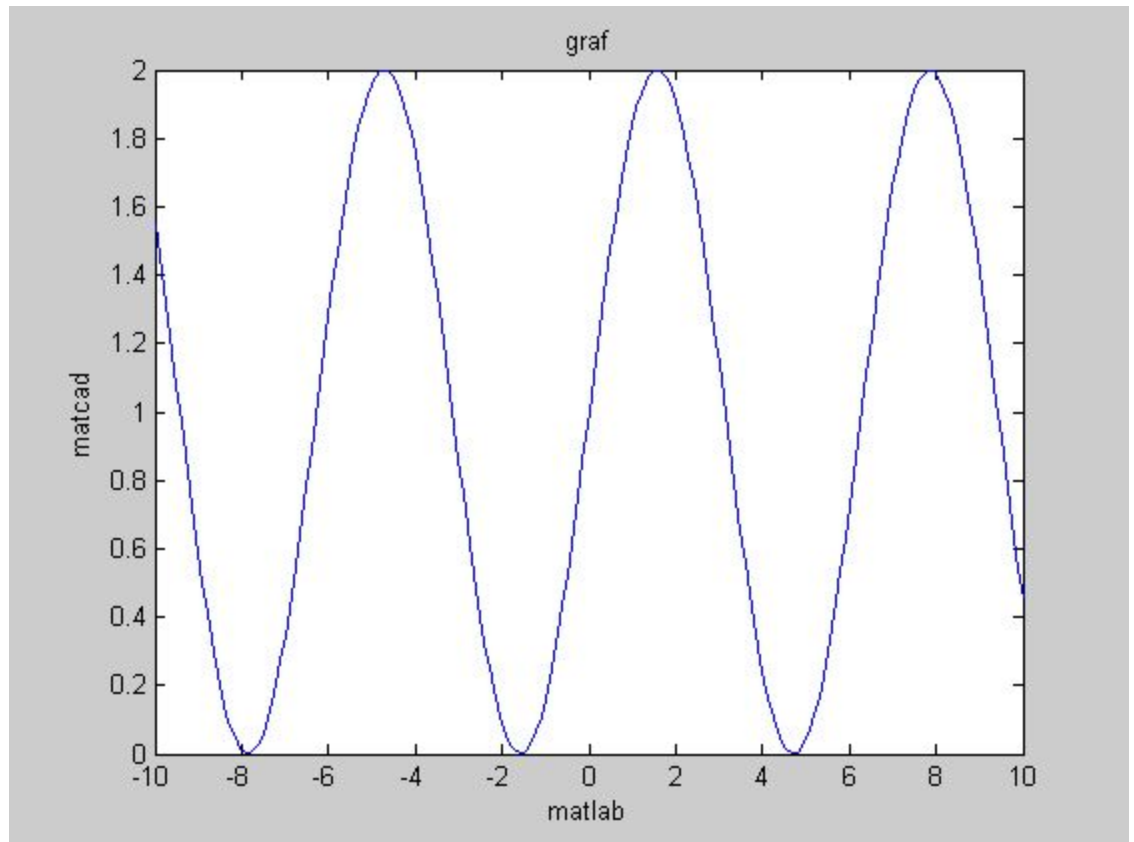
```
y=sin(x)+1;
```

```
plot(x,y)
```

```
xlabel('matlab')
```

```
ylabel('matcad')
```

```
title('graf')
```



Форматирование графиков

Названия отдельных графиков

Синтаксис команды

`legend('string1','string2',...)`

Пример 9

```
clear all
```

```
x=-10:0.1:10;
```

```
y1=sin(x)+1;
```

```
y2=sin(x);
```

```
y3=cos(x);
```

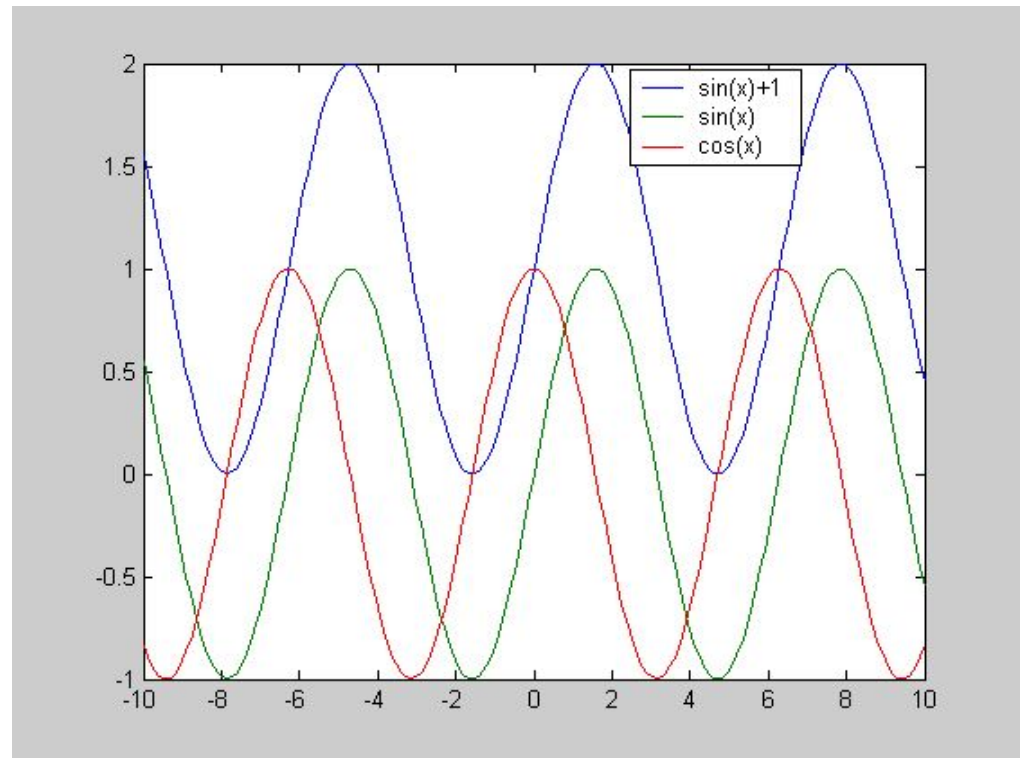
```
xlabel('matlab')
```

```
ylabel('matcad')
```

```
title('graf')
```

```
plot(x,y1,x,y2,x,y3)
```

```
legend('sin(x)+1','sin(x)','cos(x)')
```





Задание на самостоятельную работу

1. С помощью каких команд задается требуемый цвет графика?
2. С помощью каких команд задается требуемый тип линии на графике?
3. С помощью каких команд осуществляется масштабирования осей?
4. С помощью каких команд осуществляется введение надписей на графиках?
5. С помощью каких команд осуществляется введение линий уровня по различным осям?



2. Столбцовые диаграммы

В прикладных расчетах часто встречаются графики, именуемые столбцовыми диаграммами, отражающие содержание некоторого вектора X . При этом каждый элемент вектора представляется столбцом, высота которого пропорциональна значению элемента. Столбцы нумеруются и масштабируются по отношению к максимальному значению наиболее высокого столбца. Выполняет построение такого графика команда `bar(X)`.

Столбцовые диаграммы — лишь один из многих типов *графиков*, которые может строить система MATLAB.

Команды построения столбцовых диаграмм

Основные команды `bar`, `barh`

Синтаксис команд

`bar(Y)`

`bar(x, Y)`

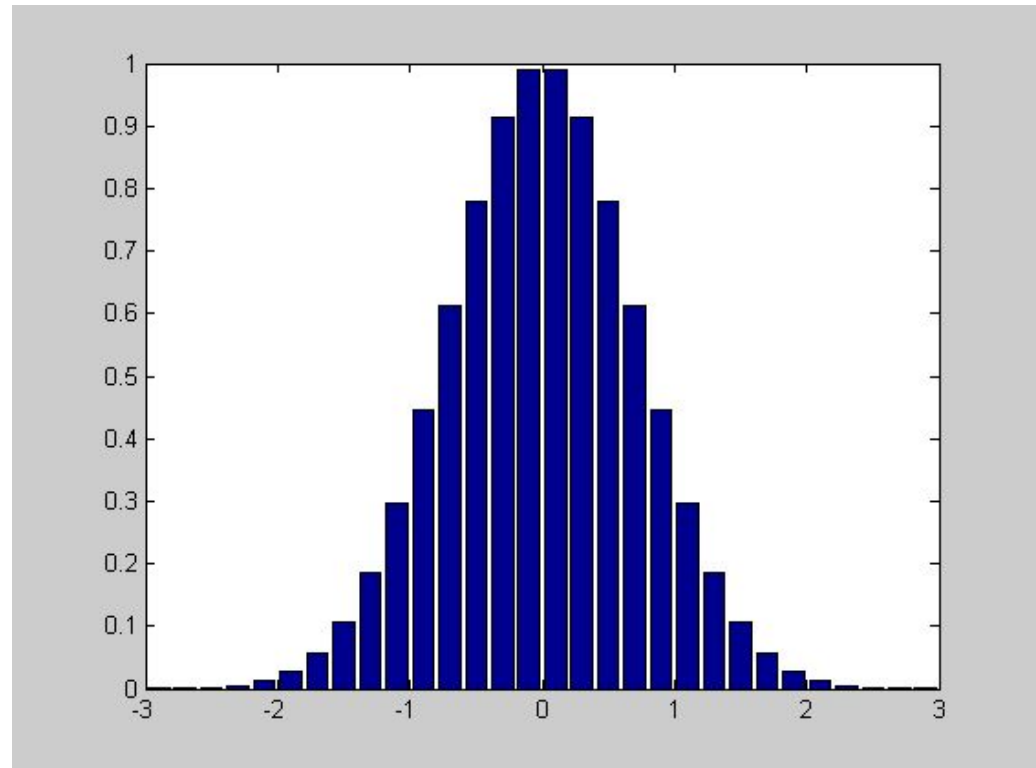
`bar(..., width)`

Пример 10

```
clear all;
```

```
x = -2.9:0.2:2.9;
```

```
bar(x, exp(-x.*x))
```



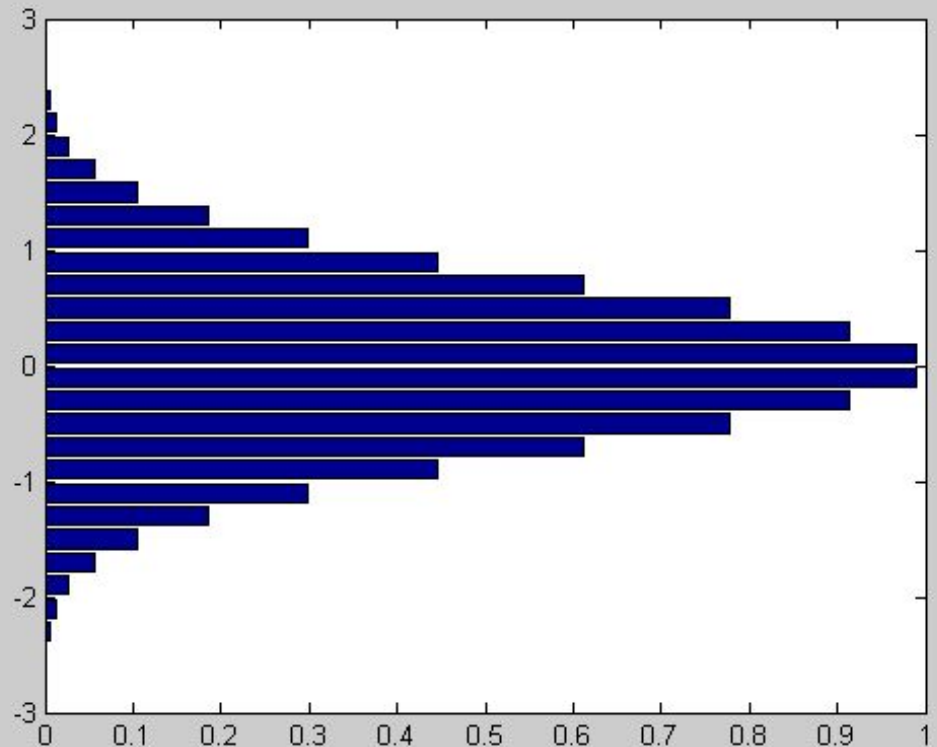
Команды построения столбцовых диаграмм

Пример 11

```
clear all;
```

```
x = -2.9:0.2:2.9;
```

```
barh(x,exp(-x.*x))
```

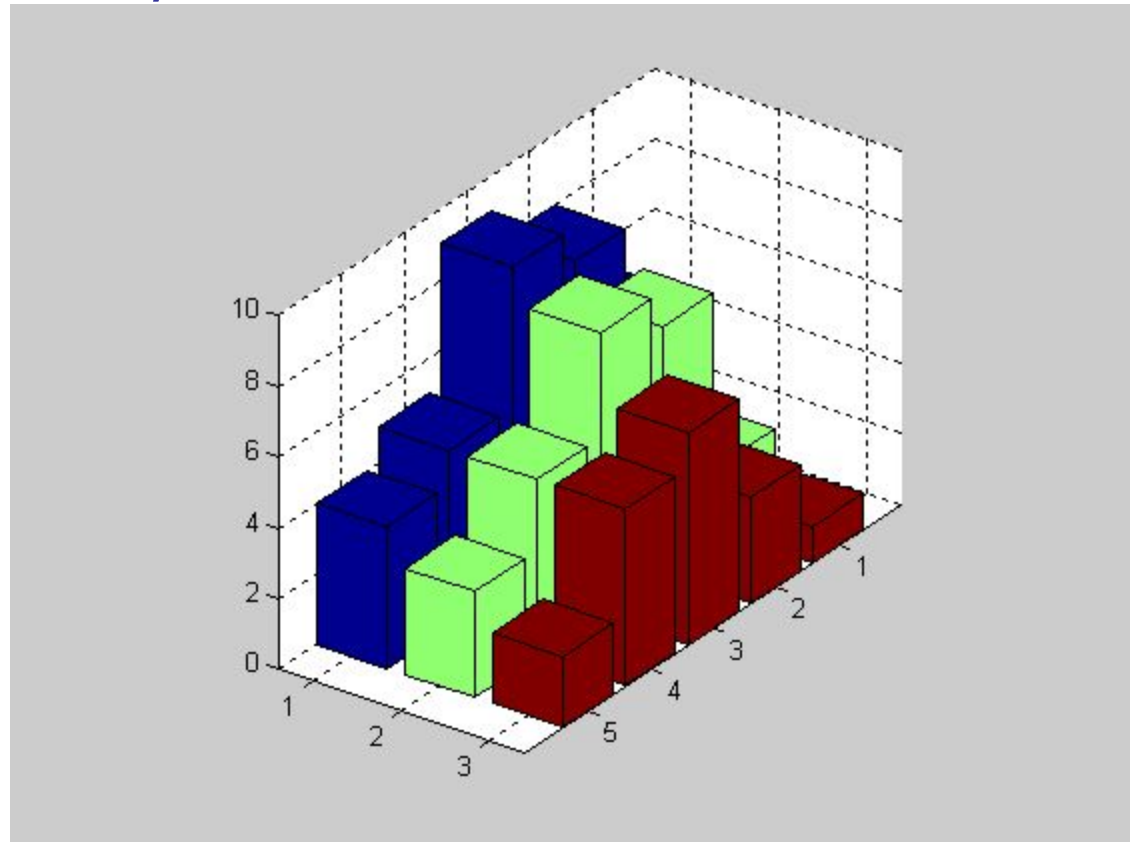


Команды построения трехмерных столбцовых диаграмм

Синтаксис команд `bar3`, `barh3`

Пример 12

```
clear all;  
Y = [5 2 1  
      8 7 3  
      9 8 6  
      5 5 5  
      4 3 2];  
bar3(Y)
```





3. Построение трехмерных графиков

Синтаксис команд `mesh`, `meshz`, `meshc`

`mesh(X,Y,Z)`

`mesh(Z)`

`mesh(...,C)`

(X – вектор $(1:n)$; Y - вектор $(1:m)$, тогда $[m,n]$ = размерность матрицы Z)

Примеры построения трехмерных графиков

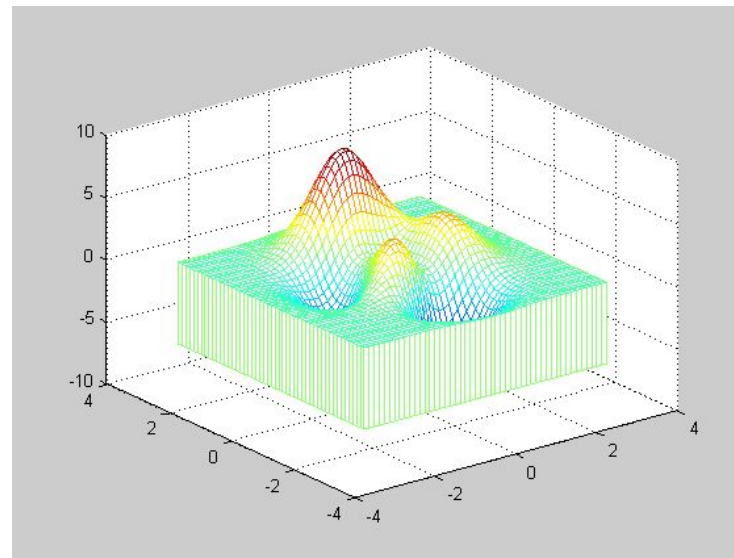
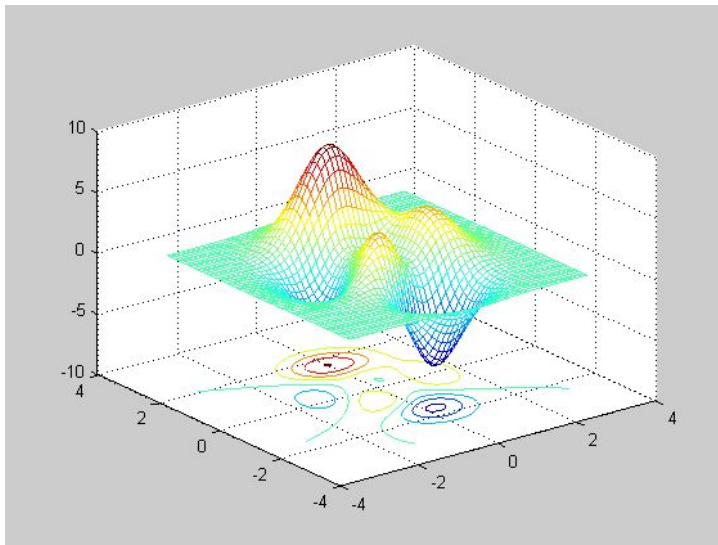
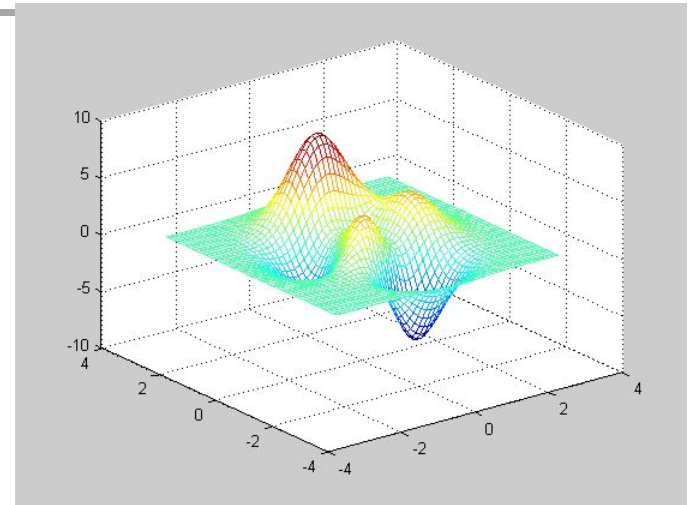
Пример 13

```
clear all;
```

```
[X,Y] = meshgrid(-3:.125:3);
```

```
Z = peaks(X,Y);
```

```
meshc(X,Y,Z);
```



Примеры построения трехмерных графиков

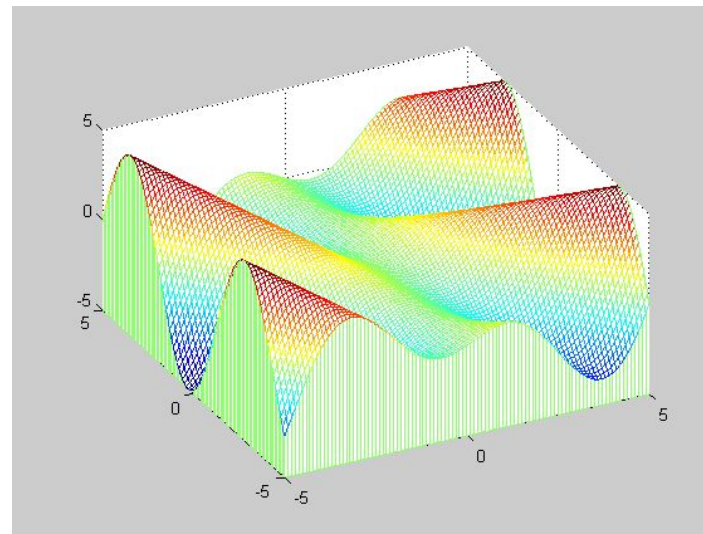
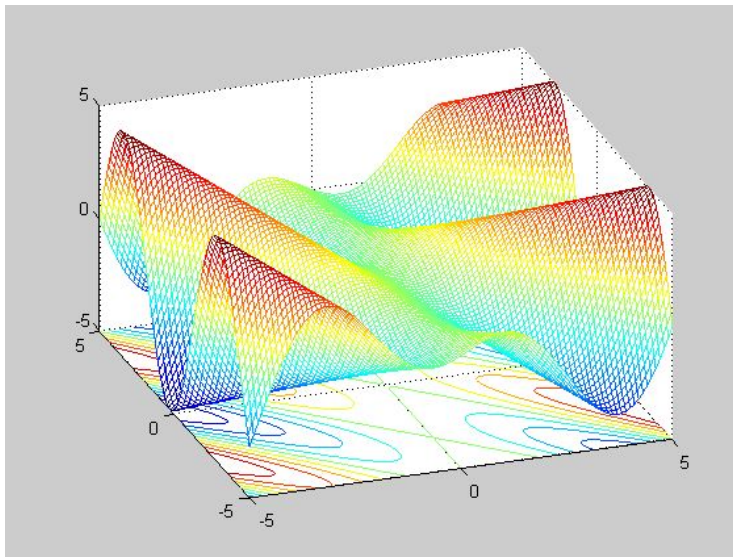
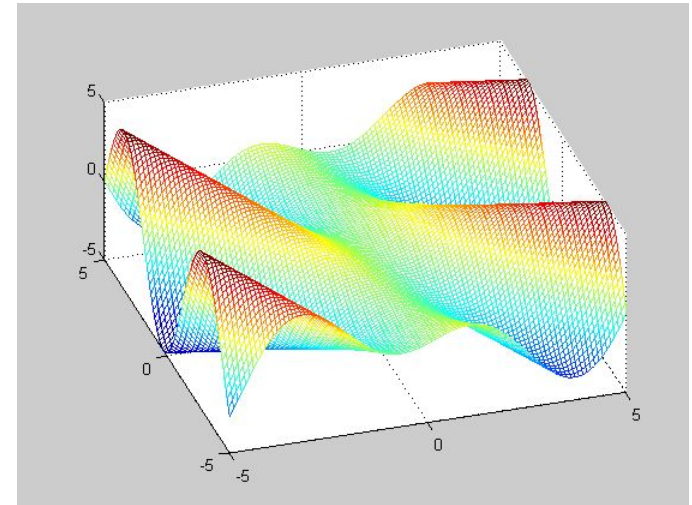
Пример 14

```
clear all;
```

```
[X,Y]=meshgrid(-5:0.1:5);
```

```
Z=X.*sin(X+Y);
```

```
mesh(X,Y,Z)
```





Выводы

Рассмотрены основные приемы форматирования графиков, в основном используя средства панели инструментов и отдельные команды из меню графического окна.

Хотя многие приемы форматирования графики заимствованы из технологии визуально-ориентированного программирования, в базовой системе MATLAB (без дополнительных пакетов расширения (toolbox)) все еще отсутствует полноценная возможность такого программирования, даже с учетом расширенных возможностей дескрипторной графики. Пользователь может лишь записать на диск копии созданных графиков в формате растрового изображения (.bmp) и использовать их в целях иллюстрации своих материалов.

Средства MATLAB позволяют опытным программистам создать расширения системы с визуально-ориентированной технологией программирования. Самым наглядным примером этого является система моделирования динамических объектов Simulink с набором моделей из готовых блоков. При этом автоматически создается не только сложная графическая блок-схема моделируемого устройства, но и система уравнений состояния, решение которой и является основой моделирования