ЛАБОРАТОРНАЯ РАБОТА 3. РАБОТА С ГРАФИКАМИ.

3.1. Цель: изучение команд построения графиков в пакете MATLAB.

3.2. Порядок выполнения работы

1. Изучите теоретическую часть. Выполните задания, соответствующие номеру Вашего варианта, и продемонстрируйте их преподавателю.

2. Оформите отчет по лабораторной работе, который должен содержать:

- титульный лист;
- цель работы;
- задание и исходные данные варианта;
- протокол сеанса работы в MATLAB.

3.3. Методические рекомендации

В пакете MATLAB имеются широкие возможности по построению и редактированию двумерных и трехмерных графиков. Графический объект имеет множество свойств. Большинство команд высокоуровневой графики, автоматически устанавливают свойства графических объектов и обеспечивают воспроизведение графиков в нужной системе координат, палитре цветов и масштабе.

Особенности графики системы MATLAB:

– Построение графика функций одной переменной;
– Столбцовые диаграммы;
– Построение трехмерных графиков;
– Вращение графиков мышью;
– Контекстное меню графиков;
– Управление форматом графиков.

Графический объект имеет множество свойств. Большинство команд построения графиков автоматически устанавливают эти свойства и обеспечивают воспроизведение графика в нужной системе координат, палитре цветов, масштабе и т.д.

Изучим некоторые команды построения двумерных графиков.

3.3.1. Построение двумерных графиков

3.3.1.1. Построение графиков отрезками прямых

Функции одной переменной $y(x)$ находят широкое применение в практике математических и других расчетов, а также в технике компьютерного математического моделирования. Для отображения таких функций используются графики в декартовой (прямоугольной) системе координат. При этом обычно строятся две оси – горизонтальная X и вертикальная Y, и задаются координаты x и y, определяющие узловые точки функции $y(x)$. Эти точки соединяются друг с другом отрезками прямых, т.е. при построении графика осуществляется линейная интерполяция для промежуточных точек. Поскольку MATLAB – матричная система, совокупность точек $y(x)$ задается векторами X и Y одинакового размера.

Команда plot служит для построения графиков функций в декартовой системе координат. Эта команда имеет ряд параметров, рассматриваемых ниже.

1) plot (X, Y) – строит график функции $y(x)$, координаты точек (x, y) которой берутся из векторов одинакового размера Y и X. Если X или Y – матрица, то строится семейство графиков по данным, содержащимся в строках матрицы.

2) plot (Y) – строит график $y(i)$, где значения y берутся из вектора Y, а i представляет собой индекс соответствующего элемента;

3) plot(X, Y, S) – аналогична команде plot(X, Y), но тип линии графика можно задавать с помощью строковой константы S.

Значениями константы S могут быть следующие символы.

Цвет линии		
Y – Желтый	R – Красный	G – Зеленый
M – Фиолетовый	K – Черный	B – Синий
C – Голубой	W – Белый	
Тип точки		**Тип линии**
. – Точка	V – Треугольник (вниз)	- – Сплошная
O – Окружность	^ – Треугольник (вверх)	: – Двойной пунктир
X – Крест	< – Треугольник (влево)	-. – Штрих-пунктир
+ – Плюс	> – Треугольник (вправо)	-- – Штриховая
* – Звездочка	P – Пятиугольник	
S – Квадрат	H – Шестиугольник	
D – Ромб		

Пример:

```
>> x=[0:0.5:10];
>> y=[sin(x); cos(x)];
>> plot(x, y(1, :), 'm+-', x, y(2, :), 'r*-')
```

Графики функций *sin*(*x*) и *cos*(*x*) приведены на рисунке 2.

3.3.1.2. Графики в логарифмическом масштабе

Для построения графиков функций со значениями *x* и *y*, изменяющимися в широких пределах, нередко используются логарифмические масштабы. Рассмотрим команды, которые используются в таких случаях.

– loglog(...) – синтаксис команды аналогичен ранее рассмотренному для функции plot(...). Логарифмический масштаб используется для координатных осей *X* и *У*.

Пример.

```
>> x=[0:1:10];
>> loglog(x,exp(x))
>> grid on
```

График функции *exp*(*x*) в логарифмическом масштабе приведен на рисунке 3.

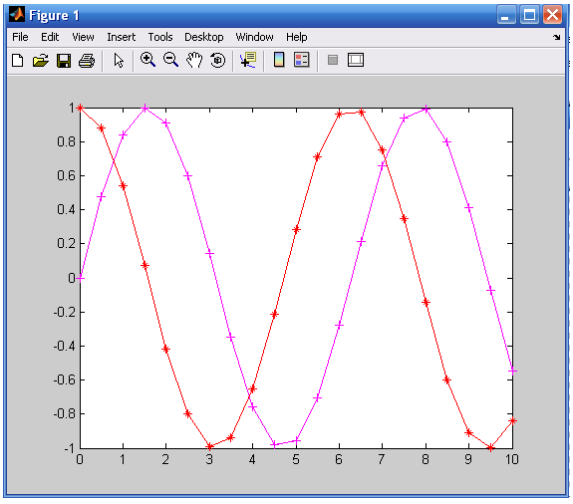


Рисунок 2 – Графики функций *sin*(*x*) и *cos*(*x*) в декартовой системе координат

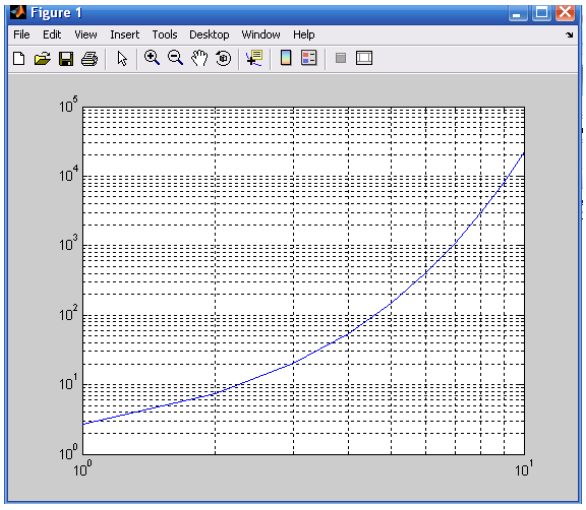


Рисунок 3 – График функции *exp*(*x*) в логарифмическом масштабе

3.3.1.3. Графики в полулогарифмическом масштабе

В некоторых случаях предпочтителен полулогарифмический масштаб графиков, когда по одной оси задается логарифмический масштаб, а по другой – линейный.

Для построения графиков функций в полулогарифмическом масштабе используются следующие команды:

1) semilogx(...) – строит график функции в логарифмическом масштабе (основание 10) по оси *X* и линейном по оси *Y;*

2) semilogy (...) – строит график функции в логарифмическом масштабе по оси *Y* и линейном по оси *X*.

Пример.

```
>> x=[0:1:10];
>> semilogy(x,exp(x))
>> grid on
```

График функции *exp*(*x*) в полулогарифмическом масштабе приведен на рисунке 4.

3.3.1.4. Столбцовые диаграммы

Столбцовые диаграммы широко используются в обработке данных, связанных с финансами и экономикой, а также в математических исследованиях. Ниже представлены команды для построения таких диаграмм:

1) bar (X, Y) – строит столбцовый график элементов вектора *Y* (или группы столбцов для матрицы *Y*) со спецификацией положения столбцов, заданной значениями элементов вектора *X*, которые должны монотонно возрастать;

2) bar (Y) – строит график значений элементов матрицы *Y* так же, как указано выше, при этом фактически используется вектор *X* =[1: *m*];

3) bar (Y,WIDTH) или BAR(Y,WIDTH) – команда аналогична ранее рассмотренным, но со спецификацией ширины столбцов (при WIDTH > 1 столбцы в одной и той же позиции перекрываются). По умолчанию задано WIDTH = 0.8.

Пример.

```
>> bar(rand(4,6))
```

График представлен на рисунке 5.

Существуют и другие типы двумерных графиков: графики с зонами погрешности, график дискретных отсчетов функции, графики в полярной системе координат, угловые гистограммы, графики векторов, график проекций векторов на плоскость, контурные графики и др.

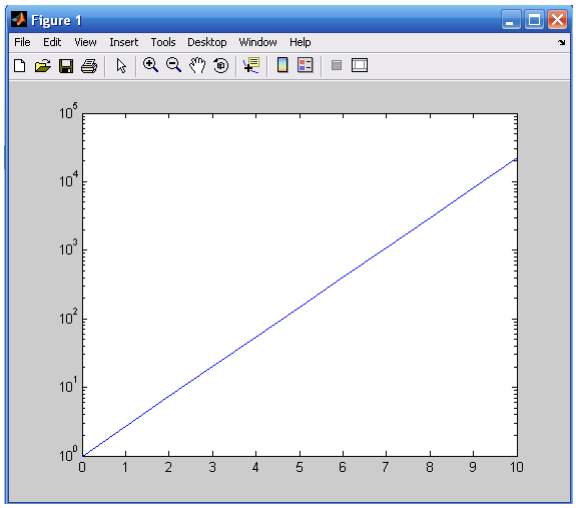


Рисунок 4 – График функции *exp*(*x*) в полулогарифмическом масштабе

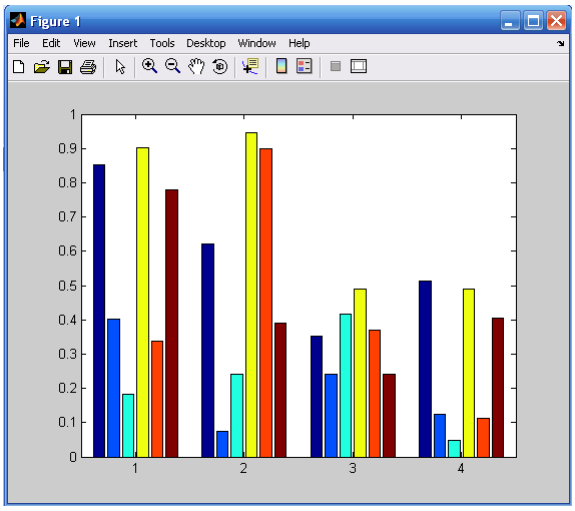


Рисунок 5 – Столбцовая диаграмма

3.3.1.5. Классическая гистограмма

Классическая *гистограмма* характеризует числа попаданий значений элементов вектора *Y* в *M* интервалов с представлением этих чисел в виде столбцовой диаграммы. Гистограммы строят следующие команды:

1) N = hist(Y) – возвращает вектор чисел попаданий для 10 интервалов, выбираемых автоматически. Если *Y* – матрица, то выдается массив данных о числе попаданий для каждого из ее столбцов;

2) N = hist(Y, M) – аналогична рассмотренной выше команде, но используется *M* интервалов (*M* – скаляр);

3) N = hist(Y, X) – возвращает числа попаданий элементов вектора *Y* в интервалы, центры которых заданы элементами вектора *X*;

4) [N, X] = HIST(...) – возвращает числа попаданий в интервалы и данные о центрах интервалов.

Пример построения гистограммы для 1000 случайных чисел:

```
>> x=-5 : 0.5 : 5
>> y=randn(1000, 1)
>> hist(y, x)
```

Гистограмма представлена на рисунке 6.

3.3.1.6 Лестничные графики

Лестничные графики визуально представляют собой ступеньки с огибающей, представленной функцией *y*(*x*). Такие графики используются, например, для отображения процессов квантования функции *y*(*x*), представленной рядом своих отсчетов. При этом в промежутках между отсчетами значения функции считаются постоянными и равными величине последнего отсчета. Для построения лестничных графиков в системе MATLAB используются следующие команды:

1) stairs(Y) – строит лестничный график по данным вектора *Y*;

2) stairs(X, Y) – строит лестничный график по данным вектора *Y* с координатами *x* переходов от ступеньки к ступеньке, заданными значениями элементов вектора *X*;

3) stairs(..., S) – аналогична по действию вышеописанным командам, но строит график линиями, стиль которых задается строками *S*.

Пример построения лестничного графика:

```
>> x=[0 : 0.25 : 10]
>> stairs(x, x.^3)
```

Лестничный график представлен на рисунке 7.

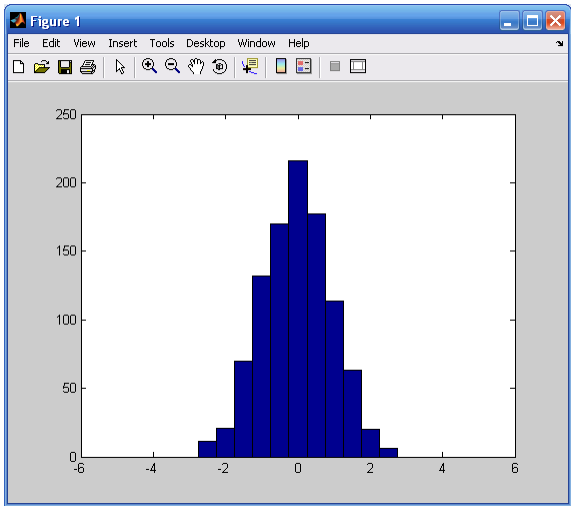


Рисунок 6 – Гистограмма

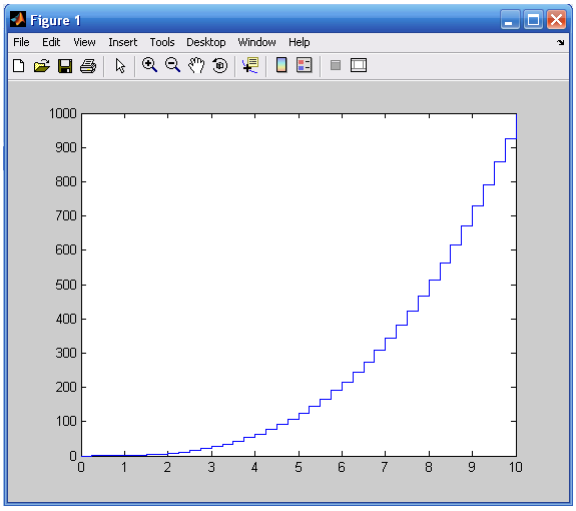


Рисунок 7 – Лестничный график

3.3.1.7. Круговые диаграммы

Команды построения *круговых диаграмм*:

1) pie(X) – строит круговую диаграмму по данным нормализованного вектора X/SUM(X), SUM(X) – сумма элементов вектора, если SUM(X) ≤ 1.0, то значения в X непосредственно определяют площадь секторов;

2) pie(X, EXPLODE) – строит круговую диаграмму, у которой отрыв секторов от центра задается вектором EXPLODE, который должен иметь тот же размер, что и вектор данных X.

Пример построения круговой диаграммы, где третий и шестой сектора отделены от остальных:

```
>> x=[1 5 10 15 20];
>> pie(x, [0 5 0 10 0])
```

Круговая диаграмма представлена на рисунке 8.

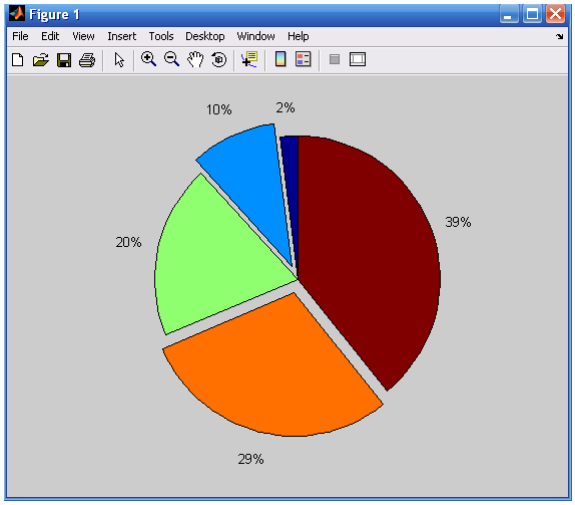


Рисунок 8 – Круговая диаграмма

3.3.2. Построение трехмерных графиков

3.3.2.1 Подготовка данных для построения трехмерных графиков

Трехмерные поверхности обычно описываются функцией двух переменных $z(x, y)$. Специфика построения трехмерных графиков требует не просто задания ряда значений x и y, то есть векторов X и Y. Она требует определения для X и Y двумерных массивов – матриц. Для создания таких массивов служит функция meshgrid.

В основном функция meshgrid используется совместно с функциями построения графиков трехмерных поверхностей. Функция meshgrid записывается в следующих формах:

1) [X,Y] = meshgrid(x) – аналогична [X,Y] = meshgrid(x, x);

2) [X, Y, Z] = meshgrid(x, y, z) – возвращает трехмерные массивы, используемые для вычисления функций трех переменных и построения трехмерных графиков;

3) [X, Y] = meshgrid(x, y) – преобразует область, заданную векторами x и y, в массивы X и Y, которые могут быть использованы для вычисления функции двух переменных и построения трехмерных графиков. Строки выходного массива X являются копиями вектора x; а столбцы Y – копиями вектора y.

Пример.

```
>> [x, y] = meshgrid(1:4, 14:17)
x =
   1   2   3   4
   1   2   3   4
   1   2   3   4
   1   2   3   4
```

```
y =
   14   14   14   14
   15   15   15   15
   16   16   16   16
   17   17   17   17
```

Функция ndgrid является многомерным аналогом функции meshgrid:

1) [X1, X2, X3,...] = ndgrid(x1, x2, x3....) – преобразует область, заданную векторами $x1$, $x2$, $x3$... в массивы $X1$, $X2$, $X3$..., которые могут быть использованы для вычисления функций нескольких переменных и многомерной интерполяции, i-я размерность выходного массива X_i является копией вектора x_i;

2) [X1, X2 ....] = ndgrid(x) – аналогична [X1, X2....] = ndgrid(x, x, ...).

3.3.2.2 Трехмерные поверхности

Команда plot3(...) является аналогом команды plot(...), но относится к функции двух переменных $z(x, y)$. Она строит аксонометрическое изображение трехмерных поверхностей и представлена следующими вариантами:

1) plot3(x, y, z) – строит массив точек, представленных векторами x, y и z, соединяя их отрезками прямых;

2) plot3(X, Y, Z) – строит точки с координатами $X(i,:)$, $Y(i,:)$ и $Z(i,:)$ и соединяет их отрезками прямых, X, Y и Z – три матрицы одинакового размера.

Пример.

```
>> [x, y]= ndgrid(-4:.25:4, -4:.25:4);
>> z=x.^2+y.^2;
>> plot3(x,y,z)
```

Построенный график приведен на рисунке 9.

3.3.2.3 Сетчатые графики

Наиболее представительными и наглядными являются сетчатые графики поверхностей с заданной или функциональной окраской. Имеются такие команды:

1) mesh(X, Y, Z, C) – выводит в графическое окно сетчатую поверхность $Z(X, Y)$ с цветами узлов поверхности, заданными массивом C;

2) mesh(X, Y, Z) – аналог предшествующей команды при $C = Z$, используется функциональная окраска, при которой цвет задается высотой поверхности.

Возможны также варианты команды mesh(x, y, Z), mesh(x, y, Z, C), mesh(Z) и mesh(Z,C).

Пример.

```
>> [x1, x2] = ndgrid(-2:.5:2, -2:.5:2);
>> z = x1.*exp(-x1.^2-x2.^2);
>> mesh(z)
```

Построенный график приведен на рисунке 10.

3.3.2.4. Закраска объемных многоугольников

Для закраски многоугольников, определенных в пространстве, служит команда fill3:

1) fill3(X, Y, Z, C) – строит закрашенный многоугольник в пространстве с данными вершин, хранящимися в векторах X, Y и Z, и цветом, заданным палитрой C;

2) fill3(X1,Y1, Z1, C1, X2, Y2, Z2, C2,...) – другой вариант построения нескольких закрашенных многоугольников в пространстве.

Пример.

```
>> x = [1 2 3 4];
>> y = [3 5 2 4];
>> z = [5 7 3 5];
>> c = [1 5 7 10];
>> fill3(x,y,z,c)
```

Построенный график приведен на рисунке 11.

Объемные круговые диаграммы строятся командой pie3:

pie3(...) – построение трехмерной круговой диаграммы, аналогична команде pie(...).

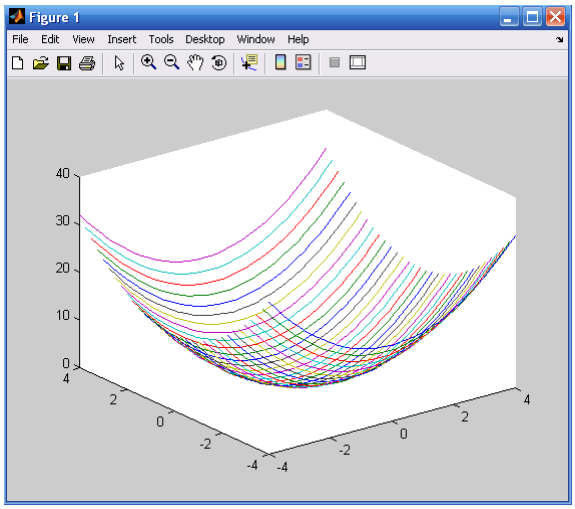


Рисунок 9 – График поверхности, построенной линиями

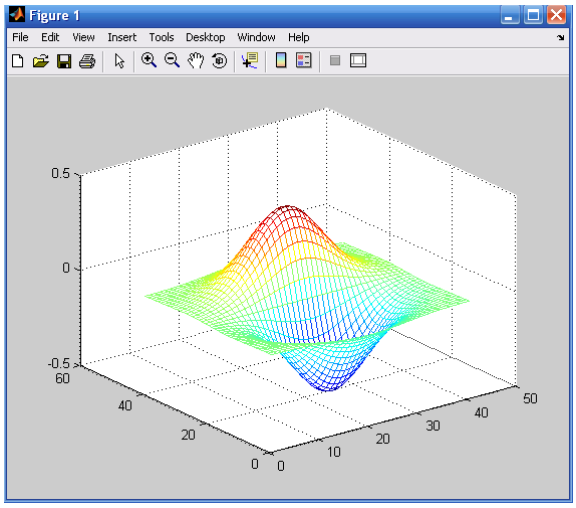


Рисунок 10 – Сетчатый график

3.3.2.5. Построение цилиндра и сферы

Для построения цилиндра и сферы в виде трехмерной фигуры с помощью команд surf (X, Y, Z) или surfl (X,Y,Z) служат команды:

1) [X,Y,Z]=cylinder(R,N) – создает массивы *X*, *Y* и *Z*, описывающие цилиндрическую поверхность с радиусом *R* и числом узловых точек *N* для последующего построения цилиндра с помощью функции surf(X, Y, Z);

2) [X,Y,Z]=cylinder(R) и [X,Y,Z]=cylinder – подобны предшествующей функции для N=20 и R = [1 1];

3) [X,Y,Z]=sphere(N) – генерирует матрицы *X*, *Y* и *Z* размера $(N+1)\times(N+1)$ для построения сферы;

4) [X,Y,Z]=sphere – аналогична предшествующей функции при $N = 20$.

Пример.

```
>> [x, y, z] = cylinder(20, 50);
>> surf(x,y,z,x)
```

Построенный цилиндр приведен на рисунке 12.

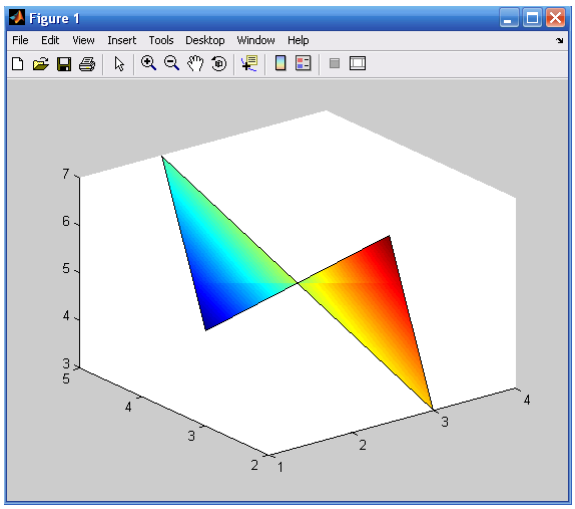


Рисунок 11 – Многоугольники

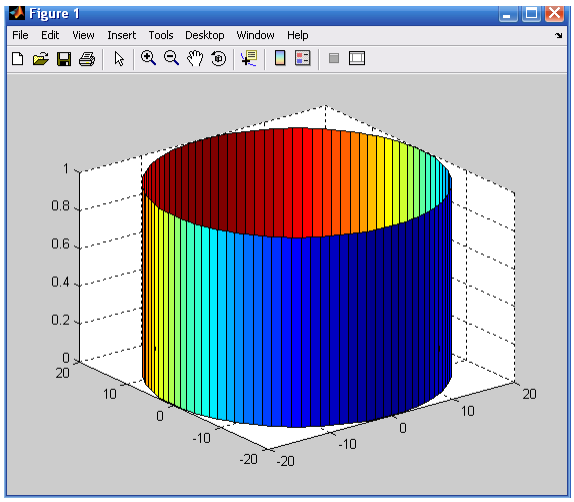


Рисунок 12 – Цилиндр

3.3.2.6. Построение объемных фигур с помощью плоских треугольников

К числу специальных видов графики относится построение объемных фигур с помощью плоских треугольников. Для построения таких фигур в виде каркаса (без окраски и отображения плоскостей) используются команды:

1) trimesh(TRI, X, Y, Z, C) – построение объемной каркасной фигуры с треугольниками, специфицированными матрицей поверхности TRI, каждая строка которой содержит три элемента и задает одну треугольную грань путем указания индексов, по которым координаты выбираются из векторов *X*, *Y*, *Z*, цвета ребер задаются вектором *C*;

2) trimesh(TRI, X, Y, Z) – построение, аналогичное предшествующему при $C = Z$, т. е. с цветом ребер, зависящим от значений высоты.

Пример построения случайной объемной фигуры, параметры которой задаются с помощью генератора случайных чисел:

```
>> x = rand(1,30);
>> y = rand(1,30);
>> z = sin(x.*y);
>> tri = delaunay(x,y);
>> trimesh(tri,x,y,z)
```

Построенная фигура приведена на рисунке 13.

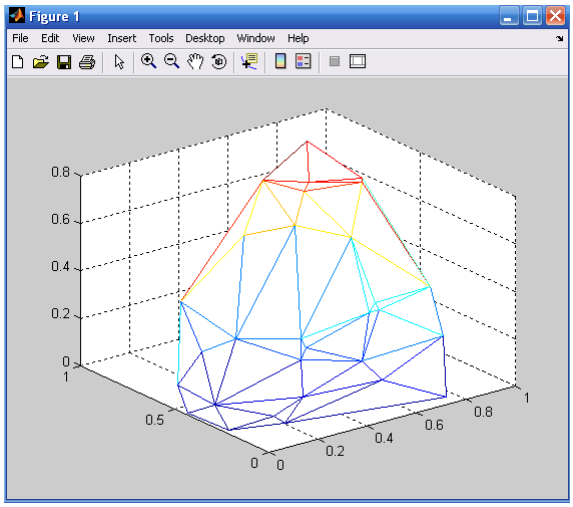


Рисунок 13 – Объемная фигура

3.3. Редактирование графиков

Для переключения в режим редактирования графика нужно щелкнуть на кнопке Edit Plot (Редактировать график) с изображением курсора-стрелки. В этом режиме графиком можно управлять с помощью контекстного меню, вызываемого щелчком правой кнопки мыши. Графики в пакете MATLAB строятся достаточно просто, так как многие свойства графиков установлены по умолчанию. К таким свойствам относятся вывод или скрытие координатных осей, положение их центра, цвет линии графика, ее толщина и т. д. Свойства и вид графиков можно менять в широких пределах с помощью параметров команд графики. Однако этот путь требует хорошего знания деталей языка программирования и дескрипторной графики MATLAB. Форматирование графиков стало более строгим и удобным: используются команды Figure Properties (свойства фигуры) и Axis Properties (свойства осей) со всеми необходимыми настройками.

Поскольку графика MATLAB обеспечивает получение цветных изображений, в ней есть ряд команд для управления цветом и различными световыми эффектами. Среди них важное место занимает установка палитры цветов. Палитра цветов RGB задается матрицей MAP из

трех столбцов, определяющих значения интенсивности красного (red), зеленого (green) и синего (blue) цветов. Их интенсивность задается в относительных единицах от 0.0 до 1.0. Например, [0 0 0] задает черный цвет, [1 1 1] – белый цвет, [0 0 1] – синий цвет. При изменении интенсивности цветов в указанных пределах возможно задание любого цвета. Таким образом, цвет соответствует общепринятому формату RGB.

Рассмотрим команды редактирования графика:

1) title('string') – установка на двумерных и трехмерных графиках титульной надписи, заданной строковой константой *'string'*;

2) xlabel('String'), ylabel('String'), zlabel('String') – команды установки надписей возле осей *x, y* и *z*;

3) text(X,Y, 'string') – добавляет в двумерный график текст, заданный строковой константой *'string'*, так что начало текста расположено в точке с координатами (*X*, *Y*), если *X* и *Y* заданы как одномерные массивы, то надпись помещается во все позиции [*x*(*i*), *y*(*i*)];

4) text(X,Y, Z, 'string') – добавляет в трехмерный график текст;

5) gtext('string') – задает выводимый на график текст в виде строковой константы *'string'* и выводит на график перемещаемый мышью маркер в виде крестика; установив маркер в нужное место, достаточно щелкнуть любой кнопкой мыши для вывода текста;

6) gtext(C) – позволяет аналогичным образом разместить многострочную надпись из массива строковых переменных *C*;

7) legend(stringl, string2, strings,...) – добавляет к текущему графику легенду в виде строк, указанных в списке параметров;

8) legend (M) – размещает легенду, используя данные из матрицы *M*;

9) legend OFF – устраняет ранее выведенную легенду;

10) axis([XMIN XMAX YMIN YMAX]) – установка диапазонов координат по осям *x* и *y* для текущего двумерного графика;

11) axis([XMIN XMAX YMIN YMAX ZMIN ZMAX]) – установка диапазонов координат по осям *x, y* и *z* текущего трехмерного графика;

12) axis auto – установка параметров осей по умолчанию;

13) grid on – добавляет сетку к текущему графику;

14) grid off – отключает сетку;

15) zoom – переключает состояние режима интерактивного изменения масштаба для текущего графика;

16) zoom (FACTOR) – устанавливает масштаб в соответствии с коэффициентом *FACTOR*;

17) zoom on – включает режим интерактивного изменения масштаба для текущего графика;

18) zoom off – выключает режим интерактивного изменения масштаба для текущего графика;

19) colormap('default') – устанавливает палитру по умолчанию, при которой распределение цветов соответствует радуге;

20) colormap(MAP) – устанавливает палитру RGB, заданную матрицей MAP;

21) shading flat – задает окраску ячеек или граней в зависимости от текущих данных;

22) shading interp – задает окраску с билинейной интерполяцией цветов;

23) shading faceted – равномерная раскраска ячеек поверхности (принята по умолчанию).

Пример установки титульной надписи и надписей по осям графиков:

```
>> [X,Y]=meshgrid([-2 : 0.1 : 2]);
>> Z=cos(X)./(X.^2+Y.^2+0.2);
>> surfl(X,Y,Z)
>> colorbar
>> colormap(gray)
```

```
>> shading interp
>> xlabel('Axis X')
>> ylabel('Axis Y')
>> zlabel('Axis Z')
>> title('Graphic')
```

Построенная фигура со всеми надписями приведена на рисунке 14.

Пример размещения надписи под кривой графика в позиции (–5, 0.7):

```
>> x=[-10:0.1:10];
>> plot(x, cos(x).^3)
>> text(-5,0.7, 'Graphic cos(x)^3')
```

Построенный график с надписью приведен на рисунке 15.

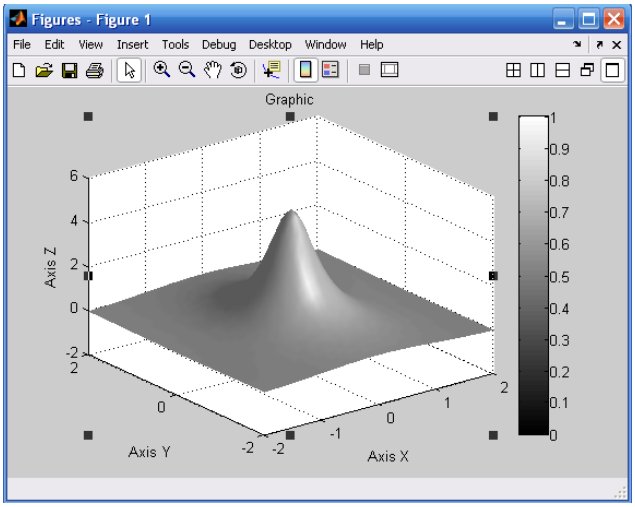


Рисунок 14 – График с надписями

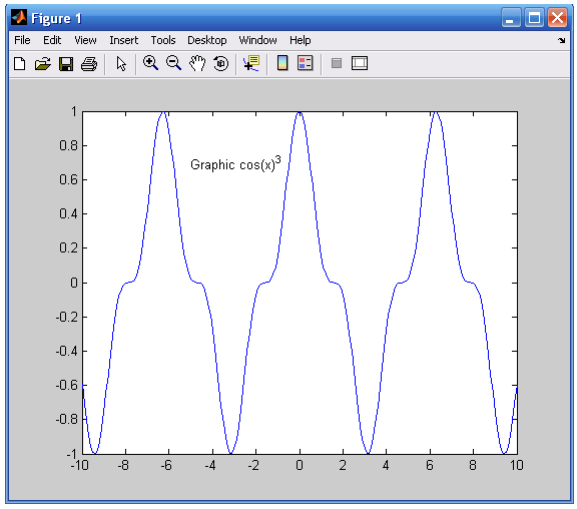


Рисунок 15 – График с надписью в определенной позиции

3.4. Задания:

<u>Задание 1.</u>

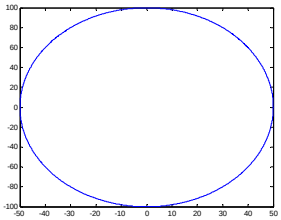
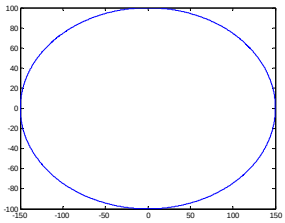
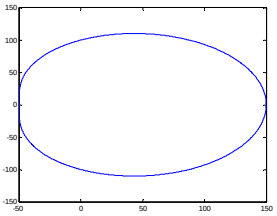
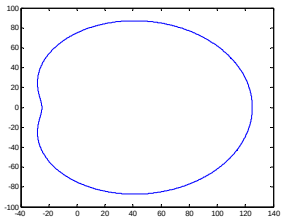
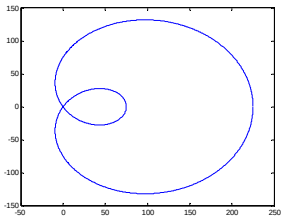
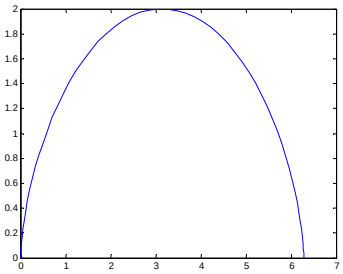
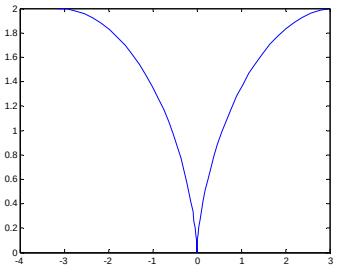
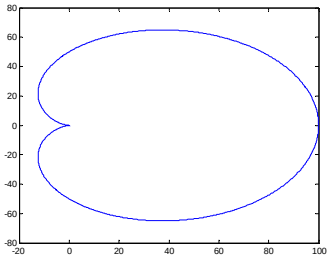
а) Построить график функции, используя команду plot. Самостоятельно задать шаг и диапазон изменения аргумента.

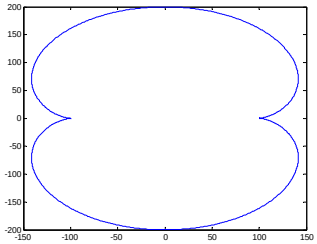
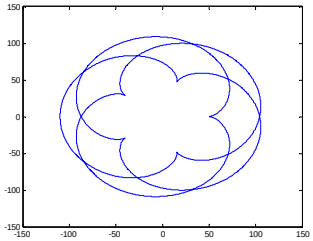
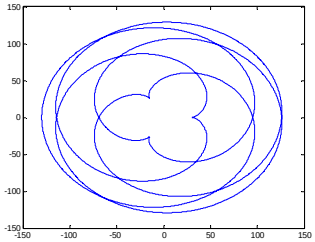
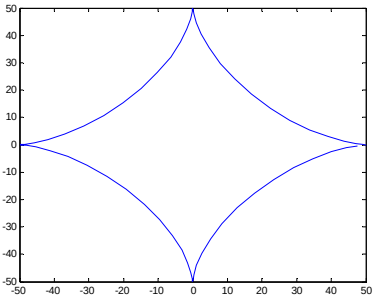
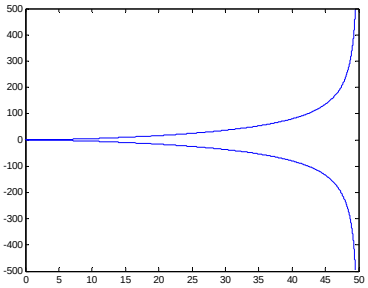
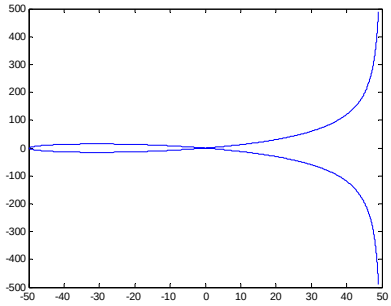
Варианты заданий

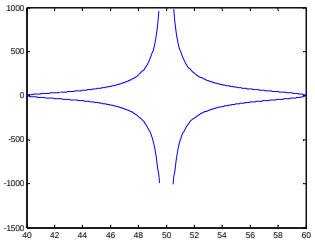
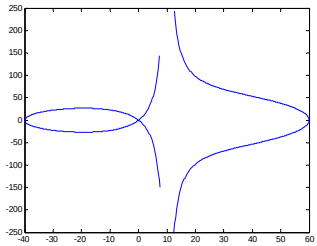
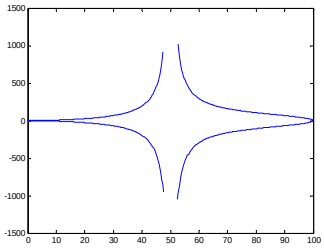
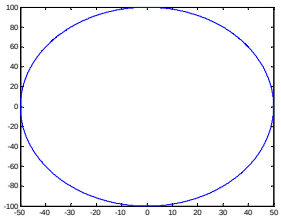
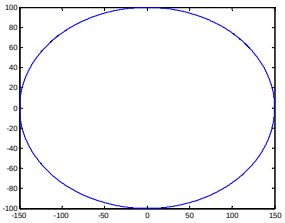
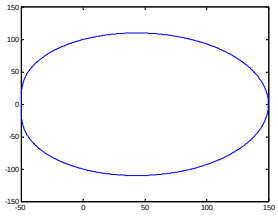
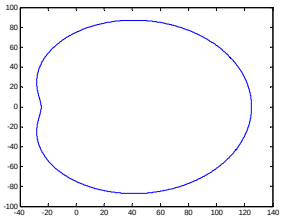
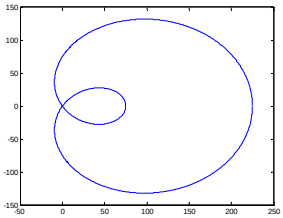
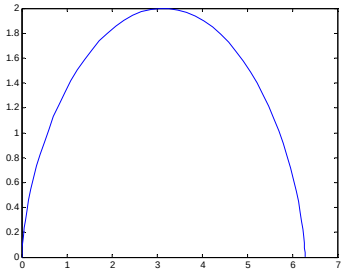
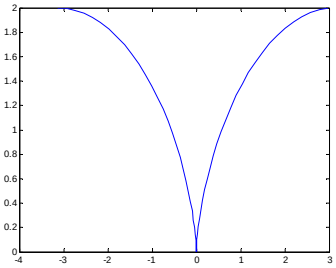
Вариант	Функция	Вариант	Функция 1	Вариант	Функция 1
1	$y = x^2 - 4$	2	$y = x^4 + 9$	3	$y = x^3 - 2x^2$
4	$y = x^2 + 5$	5	$y = x^4 - 10$	6	$y = x^3 + 4x^2$
7	$y = x^2$	8	$y = x^4 + 9x^3$	9	$y = x^4 + 2x^3 + 3$
10	$y = 1 - x^2$	11	$y = x^4 - 3x^3$	12	$y = 2x^3 - 6x^2 + 3$
13	$y = x^2 - 6x$	14	$y = x^4 - 3x^3 + 4x^2$	15	$y = x^5 - 2x^2$
16	$y = x^2 + 7x$	17	$y = x^4 + 5x^3 - 2x^2$	18	$y = x^5 + 4x^2$
19	$y = x^3 + 7$	20	$y = x^4 + 2x^2$	21	$y = x^5 - 3x^3$
22	$y = x^3 - 2$	23	$y = x^4 - 4x^2$	24	$y = x^5 + 4x^4$
25	$y = x^3 - 3x$	26	$y = x^4 - 3x^3 + 4x^2 - 4$	27	$y = x^5 - 2x^4$
28	$y = x^3 + 5x$	29	$y = x^4 + 2x^3 - 6x^2 + 3$	30	$y = x^5 + 4x^3$

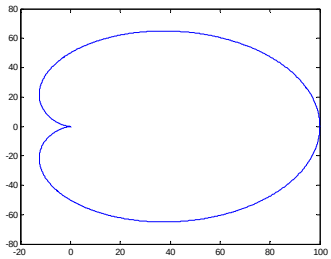
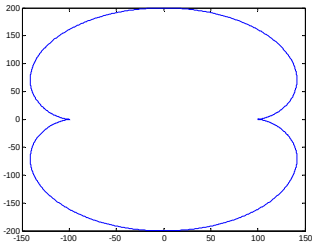
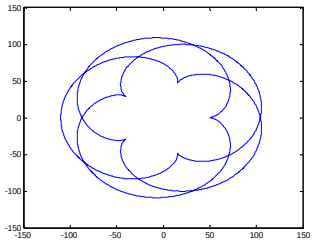
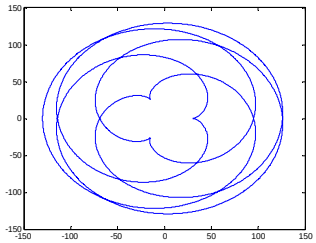
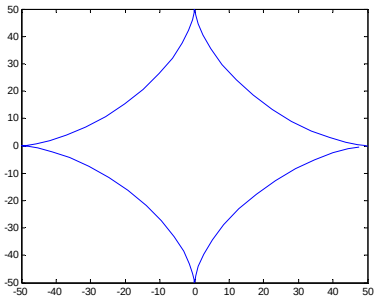
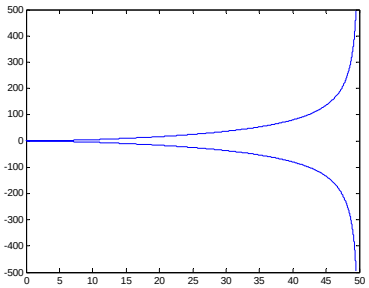
б) Построить кривую по заданному параметрическому представлению (параметрическое представление кривой l на плоскости с координатами x, y – это две функции $x = x(t)$, $y = y(t)$, определенные на одном и том же числовом множестве) (команда plot).

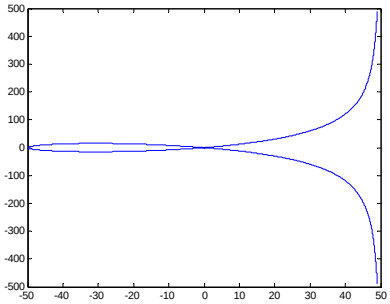
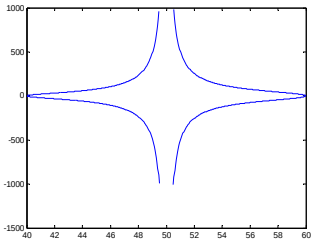
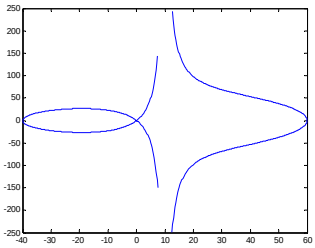
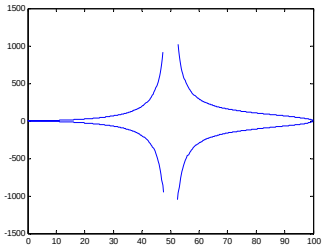
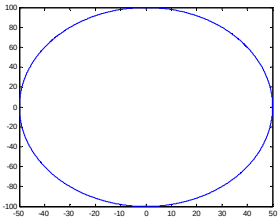
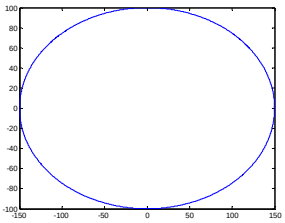
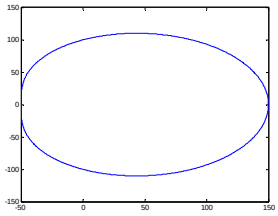
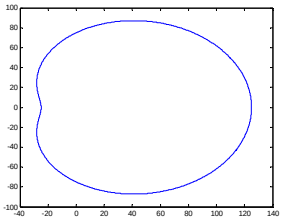
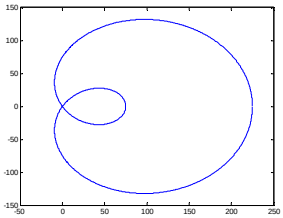
Варианты заданий

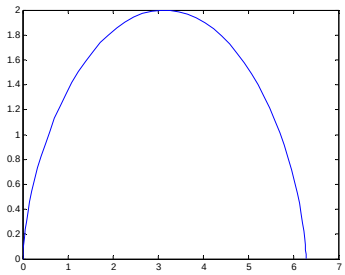
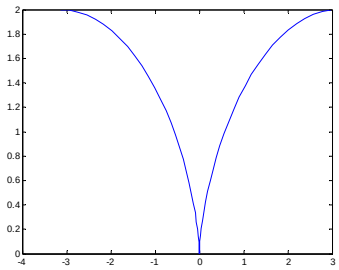
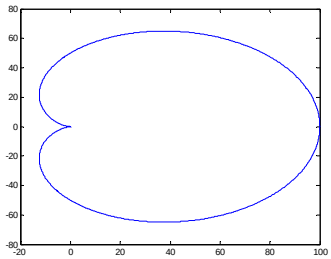
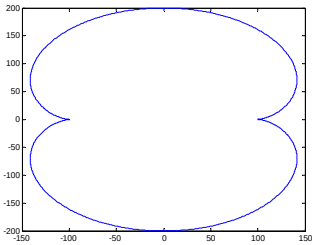
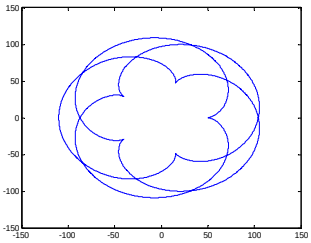
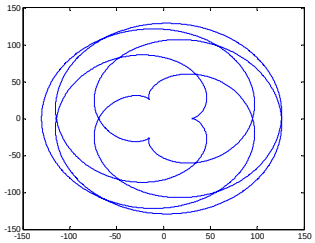
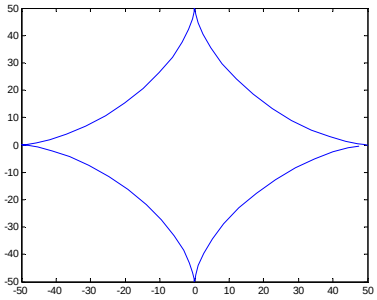
Вариант	Кривая
1	Окружность радиуса r с центром в начале координат: $x = r\cos(t)$, $y = r\sin(t)$, $t \in [0,2\pi)$.
2	Эллипс с большой и малой полуосями, равными соответственно r_1 и r_2 и расположенными параллельно осям координат: $x = r_1\cos(t)$, $y = r_2\sin(t)$, $t \in [0,2\pi)$.   $r_1 < r_2$ $r_1 > r_2$
3	Улитка Паскаля: $x = a\cos^2 t + b\cos t$, $y = a\cos t\sin t + b\sin t$, $a > 0, b > 0$, $t \in [0,2\pi)$. Рассмотреть случаи, когда $b \ge 2a$, $a < b < 2a$, $a > b$.    $b \ge 2a$ $a < b < 2a$ $a > b$
4	$x = t - \sin(t)$, $y = 1 - \cos(t)$, $t \in [-\pi,\pi]$ и $t \in [0,2\pi]$.   $t \in [0,2\pi]$ $t \in [-\pi,\pi]$
5	Кардиоида: $x = a\cos t(1+\cos t)$, $y = a\sin t(1+\cos t)$, $a > 0$, $t \in [0,2\pi)$. 

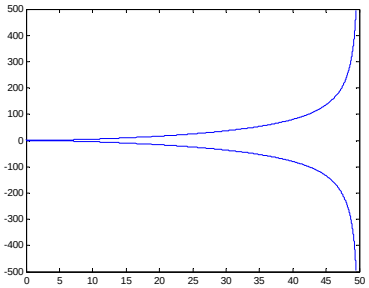
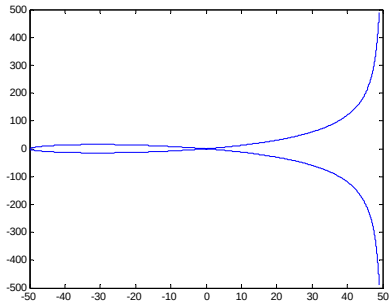
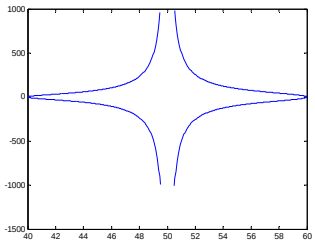
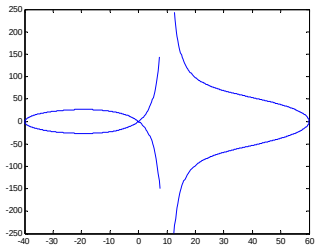
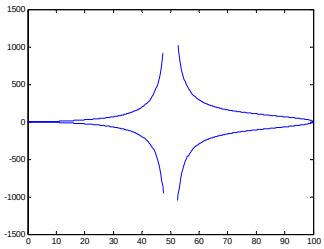
Вари-ант	**Кривая**
6	Эпициклоида: $x=(a+b)\cos t-a\cos((a+b)t/a)$, $x=(a+b)\sin t-a\sin((a+b)t/a)$, $a>0, b>0$. Рассмотреть следующие случаи: если b/a есть целое положительное число, $t\in[0,2\pi)$; если $b/a=p/q$, где p и q – положительные целые взаимно простые числа, $t\in[0,2q\pi)$.  b/a есть целое положительное число, $t\in[0,2\pi)$  если $b/a=p/q$, $a<b$, где p и q – положительные целые взаимно простые числа, $t\in[0,2q\pi)$.  если $b/a=p/q$, $a>b$, г p и q – положительные лые взаимно простые чис $t\in[0,2q\pi)$.
7	Астроида: $x=b\cos^3 t$, $y=b\sin^3 t$, $t\in[0,2\pi)$. 
8	Циссоида: $x=at^2/(1+t^2)$, $y=at^3/(1+t^2)$, $t\in(-\infty,\infty)$, $a>0$. 
9	Строфоида: $x=a(t^2-1)/(t^2+1)$, $y=at(t^2-1)/(t^2+1)$, $t\in(-\infty,\infty)$, $a>0$. 

Вариант	Кривая
10	Конхоида Никомеда: $x = a + l\cos t$, $y = atgt + l\sin t$, $t \in (-\pi/2, \pi/2)$ – правая ветвь, $t \in (\pi/2, 3\pi/2)$ – левая ветвь, $a > 0$, $l > 0$. Рассмотреть случаи, когда $l < a$, $l > a$, $l = a$.    $l < a$ $l > a$ $l = a$
11	Окружность радиуса r с центром в начале координат: $x = r\cos(t)$, $y = r\sin(t)$, $t \in [0, 2\pi)$.
12	Эллипс с большой и малой полуосями, равными соответственно r_1 и r_2 и расположенными параллельно осям координат: $x = r_1\cos(t)$, $y = r_2\sin(t)$, $t \in [0, 2\pi)$.   $r_1 < r_2$ $r_1 > r_2$
13	Улитка Паскаля: $x = a\cos^2 t + b\cos t$, $y = a\cos t\sin t + b\sin t$, $a > 0, b > 0$, $t \in [0, 2\pi)$. Рассмотреть случаи, когда $b \geq 2a$, $a < b < 2a$, $a > b$.    $b \geq 2a$ $a < b < 2a$ $a > b$
14	$x = t - \sin(t)$, $y = 1 - \cos(t)$, $t \in [-\pi, \pi]$ и $t \in [0, 2\pi]$.   $t \in [0, 2\pi]$ $t \in [-\pi, \pi]$

Вари-ант	**Кривая**
15	Кардиоида: $x = a\cos t(1+\cos t)$, $y = a\sin t(1+\cos t)$, $a > 0$, $t \in [0,2\pi)$. 
16	Эпициклоида: $x = (a+b)\cos t - a\cos((a+b)t/a)$, $x = (a+b)\sin t - a\sin((a+b)t/a)$, $a > 0, b > 0$. Рассмотреть следующие случаи: если b/a есть целое положительное число, $t \in [0,2\pi)$; если $b/a = p/q$, где p и q – положительные целые взаимно простые числа, $t \in [0,2q\pi)$.  b/a есть целое положительное число, $t \in [0,2\pi)$  если $b/a = p/q$, $a < b$, где p и q – положительные целые взаимно простые числа, $t \in [0,2q\pi)$.  если $b/a = p/q$, $a > b$, гд p и q – положительные ц лые взаимно простые числ $t \in [0,2q\pi)$.
17	Астроида: $x = b\cos^3 t$, $y = b\sin^3 t$, $t \in [0,2\pi)$. 
18	Циссоида: $x = at^2/(1+t^2)$, $y = at^3/(1+t^2)$, $t \in (-\infty,\infty)$, $a > 0$. 

Вариант	Кривая
19	Строфоида: $x = a(t^2 - 1)/(t^2 + 1)$, $y = at(t^2 - 1)/(t^2 + 1)$, $t \in (-\infty, \infty)$, $a > 0$. 
20	Конхоида Никомеда: $x = a + l\cos t$, $y = atgt + l\sin t$, $t \in (-\pi/2, \pi/2)$ – правая ветвь, $t \in (\pi/2, 3\pi/2)$ – левая ветвь, $a > 0$, $l > 0$. Рассмотреть случаи, когда $l < a, l > a, l = a$.    $l < a$ $l > a$ $l = a$
21	Окружность радиуса r с центром в начале координат: $x = r\cos(t)$, $y = r\sin(t)$, $t \in [0, 2\pi)$.
22	Эллипс с большой и малой полуосями, равными соответственно r_1 и r_2 и расположенными параллельно осям координат: $x = r_1\cos(t)$, $y = r_2\sin(t)$, $t \in [0, 2\pi)$.   $r_1 < r_2$ $r_1 > r_2$
23	Улитка Паскаля: $x = a\cos^2 t + b\cos t$, $y = a\cos t\sin t + b\sin t$, $a > 0, b > 0$, $t \in [0, 2\pi)$. Рассмотреть случаи, когда $b \geq 2a$, $a < b < 2a$, $a > b$.    $b \geq 2a$ $a < b < 2a$ $a > b$

Вари-ант	**Кривая**
24	$x = t - \sin(t),\ y = 1 - \cos(t),\ t \in [-\pi, \pi]$ и $t \in [0, 2\pi]$.  $t \in [0, 2\pi]$  $t \in [-\pi, \pi]$
25	Кардиоида: $x = a\cos t(1+\cos t),\quad y = a\sin t(1+\cos t),\ a > 0,\ t \in [0, 2\pi)$. 
26	Эпициклоида: $x = (a+b)\cos t - a\cos((a+b)t/a),\quad x = (a+b)\sin t - a\sin((a+b)t/a)$, $a > 0, b > 0$. Рассмотреть следующие случаи: если b/a есть целое положительное число, $t \in [0, 2\pi)$; если $b/a = p/q$, где p и q – положительные целые взаимно простые числа, $t \in [0, 2q\pi)$.  b/a есть целое положительное число, $t \in [0, 2\pi)$  если $b/a = p/q$, $a < b$, где p и q – положительные целые взаимно простые числа, $t \in [0, 2q\pi)$.  если $b/a = p/q$, $a > b$, гд p и q – положительные ц лые взаимно простые числ $t \in [0, 2q\pi)$.
27	Астроида: $x = b\cos^3 t,\ y = b\sin^3 t$, $t \in [0, 2\pi)$. 

Вари-ант	**Кривая**
28	Циссоида: $x = at^2/(1+t^2)$, $y = at^3/(1+t^2)$, $t \in (-\infty, \infty)$, $a > 0$. 
29	Строфоида: $x = a(t^2-1)/(t^2+1)$, $y = at(t^2-1)/(t^2+1)$, $t \in (-\infty, \infty)$, $a > 0$. 
30	Конхоида Никомеда: $x = a + l\cos t$, $y = atgt + l\sin t$, $t \in (-\pi/2, \pi/2)$ – правая ветвь, $t \in (\pi/2, 3\pi/2)$ – левая ветвь, $a > 0$, $l > 0$. Рассмотреть случаи, когда $l < a$, $l > a$, $l = a$.  $l < a$  $l > a$  $l = a$

<u>Задание 2.</u> Построить столбцовую и круговую диаграммы, классическую гистограмму и лестничный график.

<u>Задание 3.</u> Построить графики трехмерных поверхностей, используя функции plot3 и mesh.

Вари-ант	**Функция 1**	**Функция 2**	**Вари-ант**	**Функция 1**	**Функция 2**
1	$z = y^2 - x^2$	$z = \cos(y)^2 - \cos(x)^2$	2	$z = y^5 + x^5$	$z = \sin^3(y) + \cos^3(x)$
3	$z = x^2 - y^2$	$z = \cos^2(y) - \cos^2(x)$	4	$z = x^5 - y^5$	$z = \cos(y)^4 - \cos(x)^4$
5	$z = x^2 + y^2$	$z = \cos^2(y) + \cos^2(x)$	6	$z = y^5 - x^5$	$z = \cos^4(y) - \cos^4(x)$
7	$z = y^3 + x^3$	$z = \cos(y)^2 + \cos(x)^2$	8	$z = y^6 + x^6$	$z = \cos(y)^4 - \sin(x)^4$
9	$z = x^3 - y^3$	$z = \cos(y)^2 + \cos^2(x)$	10	$z = x^6 - y^6$	$z = \cos^4(y) - \sin^4(x)$
11	$z = y^3 - x^3$	$z = \cos^2(y) - \cos(x)^2$	12	$z = y^6 - x^6$	$z = \sin(y)^4 - \cos(x)^4$
13	$z = y^4 + x^4$	$z = \cos(x)^2 - \cos(y)^2$	14	$z = y^7 + x^7$	$z = \sin^4(y) - \cos^4(x)$
15	$z = x^4 - y^4$	$z = \cos(y)^2 - \sin(x)^2$	16	$z = x^7 - y^7$	$z = \sin(y)^3 + \cos(x)^3$

17	$z = y^4 - x^4$	$z = \cos^2(y) - \sin^2(x)$	18	$z = y^7 - x^7$	$z = \cos(y)^5 + \sin(x)^5$
19	$z = \sqrt{\lvert x\rvert} - \sqrt{\lvert y\rvert}$	$z = \sin(y)^2 - \cos(x)^2$	20	$z = \sqrt[3]{\lvert y\rvert} - \sqrt[3]{\lvert x\rvert}$	$z = \cos^5(y) + \sin^5(x)$
21	$z = \sqrt{\lvert x\rvert} + \sqrt{\lvert y\rvert}$	$z = \sin^2(y) - \cos^2(x)$	22	$z = \sqrt[3]{\lvert x\rvert} + \sqrt[3]{\lvert y\rvert}$	$z = \sin(y)^5 + \cos(x)^5$
23	$z = \sqrt[3]{\lvert x\rvert} - \sqrt[3]{\lvert y\rvert}$	$z = \cos(y)^3 + \cos(x)^3$	24	$z = y^9 + x^9$	$z = \sin^5(y) + \cos^5(x)$
25	$z = y^8 + x^8$	$z = \cos^3(y) + \cos^3(x)$	26	$z = x^9 - y^9$	$z = \cos(y)^5 + \sin^5(x)$
27	$z = x^8 - y^8$	$z = \cos(y)^3 + \sin(x)^3$	28	$z = y^9 - x^9$	$z = \sin^5(y) + \cos(x)^5$
29	$z = y^8 - x^8$	$z = \cos^3(y) + \sin^3(x)$	30	$z = y^{10} - x^{10}$	$z = \sin^5(y) - \cos(x)^5$

Задание 4. Построить различные трехмерные графики, используя следующие команды: fill3, pie3, trimesh, sphere, cylinder.

Задание 5. Перейти в режим редактирования графиков и проработать все команды редактирования для различных графиков.

3.5. Примерный перечень вопросов на защите работы

1. Перечислите основные команды построения двумерных графиков отрезками прямых.
2. Перечислите команды построения двумерных графиков в логарифмическом масштабе.
3. Назовите команды построения столбцовых и круговых диаграмм.
4. Назовите команды построения гистограммы и лестничного графика.
5. Перечислите команды построения трехмерных графиков.
6. Назовите команды подготовки данных для построения трехмерных графиков.
7. Перечислите команды построения цилиндра и сферы.
8. Назовите команды построения сетчатых графиков.
9. Перечислите команды построения объемных фигур с помощью плоских треугольников.
10. Назовите команды редактирования графиков.