*Задание 1*. Создать программу построения графиков двух функций в одном окне.

Команда для построения графиков нескольких функций в одном окне используется с тремя параметрами: *subplot* (*i*, *j*, *n*). Здесь *i* и *j* – число подграфиков по вертикали и горизонтали, а *n* – номер графика, который нужно сделать текущим. Номер отсчитывается от левого верхнего угла построчно.

*Методика выполнения*:

1. Вызовите редактор файлов командой меню *File → New → M-file.*
2. Наберите команды:

*x* = [0:0.1:7];

*f =* exp(–*x*);

*subplot* (1, 2, 1)

*plot (x,f*)

%построение графика sin (*x*) – э*то комментарий, который начинается со знака* %

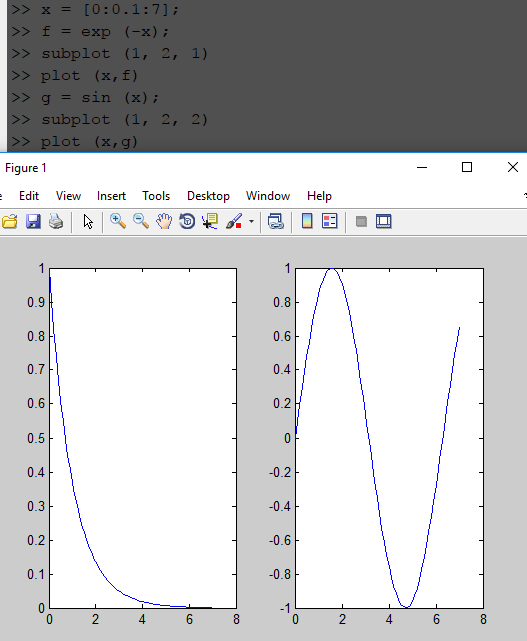
*g* = sin (*x*);

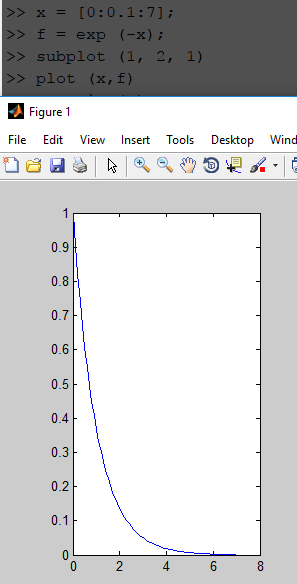
*subplot* (1, 2, 2)

*plot* (*x,g*)

1. Сохраните файл под именем *myfile.m* в подкаталоге *Work* командой *File* → *Save as*.
2. Для запуска всех команд на выполнение выберите команду меню *Debug* → *Run*. На экране появится графическое окно *Figure No.1*, содержащее графики функций.
3. Чтобы выполнить часть команд, выделите первые четыре команды листинга обычным способом и выполните их из пункта *Evaluate Selection* меню *Text*.

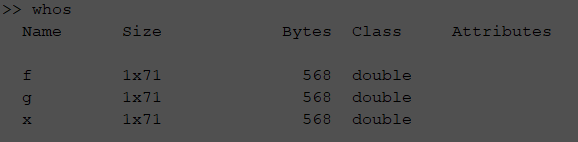
Открытие М-файла производится при помощи команды *File → Open* или командой *edit\_имя файла* из командной строки.





*Задание 2*. Выполните в редакторе М-файлов файл-про-грамму, приведенную в задании 1, и наберите команду *whos* в командной строке. Просмотрите описание переменных. Переменные, объявленные в файл-программе, можно использовать в других файл-программах.

Фактически, созданный *М-файл* становится командой, которую понимает MatLab.



*Задание 3*. Создать файл-функцию для вычисления функции.



и вычислить значение функции при *х* = 1,3.

*Методика выполнения*:

1. Откройте в редакторе новый М-файл и наберите текст листинга:

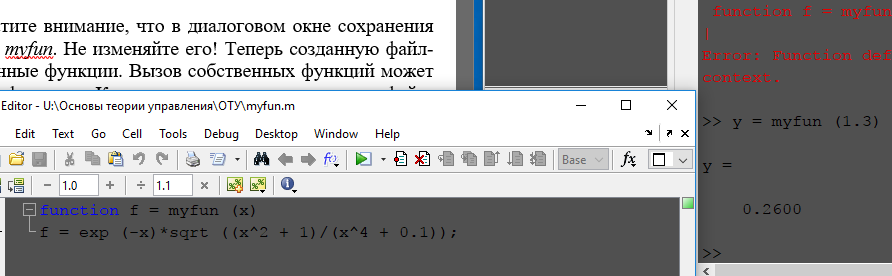
*function f = myfun* (*x*) – это команда создания файл-функции под названием *myfun*

*f =* exp(*–x*)*\**sqrt((*x*^2 + 1)/(*x*^4 + 0.1));

2. Сохраните файл в рабочем каталоге. Обратите внимание, что в диалоговом окне сохранения файла в строке *File name* уже содержится название *myfun*. Не изменяйте его! Теперь созданную файл-функцию можно использовать так же, как и встроенные функции. Вызов собственных функций может осуществляться из файл-программы или из файл-функции. Каталог, в котором содержится файл-функция, должен быть текущим или должен быть добавлен в пути поиска.

3. В командной строке вычислите значение функции

>> *y* = *myfun* (1.3)



*Задание* 4. Создать файл-функцию, вычисляющую длину радиус-вектора точки трехмерного пространства.

*Методика выполнения:*

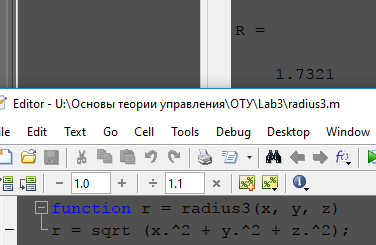
1. Откройте в редакторе М-файлов новый файл и наберите текст листинга:

*function* *r* = *radius*3(*x*, *y*, *z*)

*r* = sqrt (*x*.^2 + *y*.^2 + *z*. ^2)

2. В командной строке вычислите:

>> *R* = *radius*3 (1, 1, 1)



*Задание 5*. Создать файл-функцию, переводящую время, заданное в секундах, в часы, минуты, секунды.

*Методика выполнения*:

1. Создайте М-файл и наберите текст листинга:

*function* [*hour, minute, second*]*= hms* (*sec*)

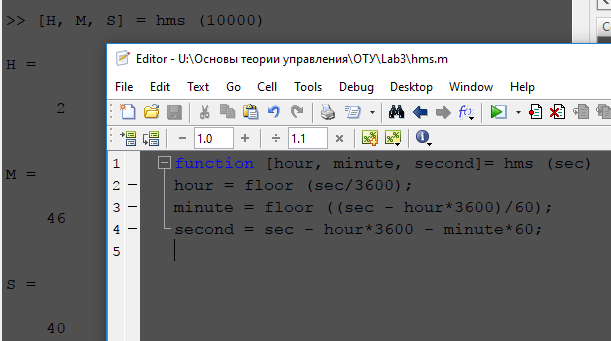
*hour = floor* (*sec/*3600);

*minute* = *floor* ((*sec* – *hour*\*3600)/60);

*second* = *sec* – *hour*\*3600 – *minute*\*60;

1. В командной строке вычислите:

>> [*H*, *M*, *S*] = *hms* (10000)



*Задание 6*. Решите на отрезке [–5, 5] уравнение:

sin *x* – *x*2 cos *x* = 0.

*Методика выполнения*:

1. Создайте файл-функцию для вычисления уравнения, назвав ее, например, *myf*.
2. Постройте график с использованием *fplot* и нанесите координатную сетку.
3. Уточните значения корней вблизи –5, –2, 0, 5.

Вычисление всех корней полинома

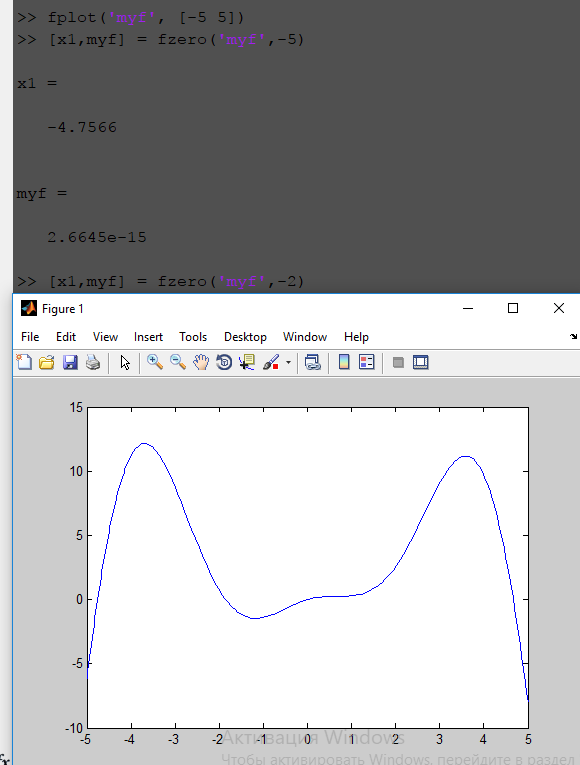
Полином в MatLab задается вектором его коэффициентов.

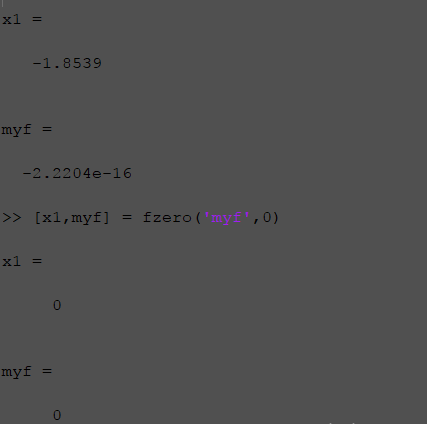
Нахождение значения полинома от некоторого аргумента осуществляется командой *polyval* (*p*, *xi*), где *p* – вектор коэффициентов, *xi* – аргумент. Число элементов вектора всегда на единицу больше его степени, нулевые коэффициенты должны содержаться в векторе.

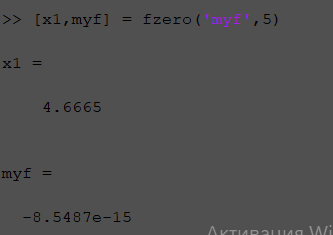
Нахождение сразу всех корней полинома осуществляется функцией *roots* (*p*).

Минимизация функции одной переменной

Поиск локальных минимумов осуществляется командой: [*xi f*] = *fminbnd* (‘*имя\_файл-функции*’, *отрезок*).



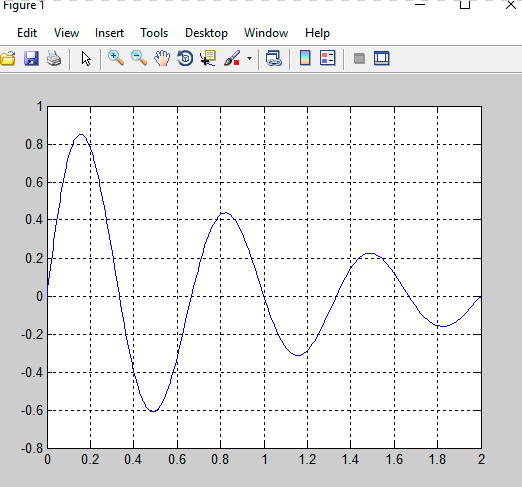


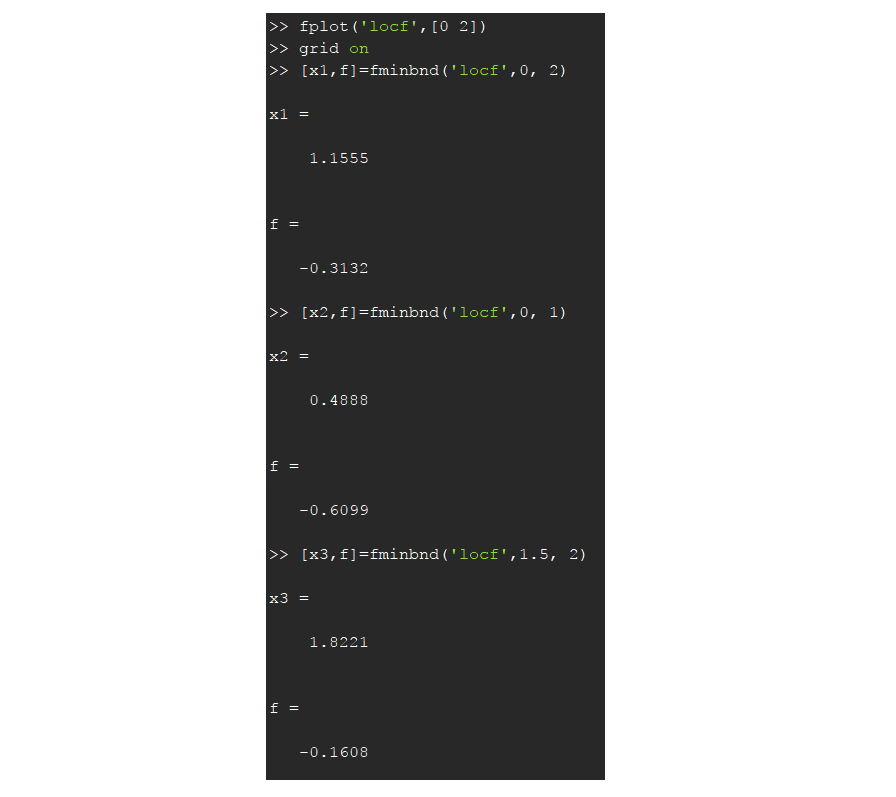


Минимизация функции одной переменной

Поиск локальных минимумов осуществляется командой: [*xi f*] = *fminbnd* (‘*имя\_файл-функции*’, *отрезок*).

*Задание 7*. Найдите локальные минимумы функции  на отрезке [0, 2].



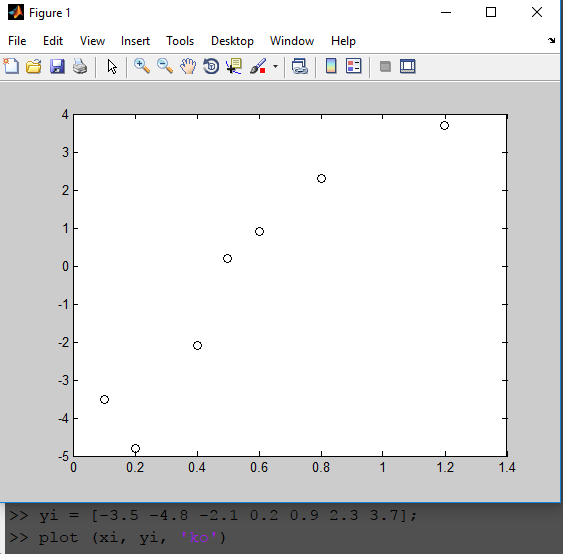
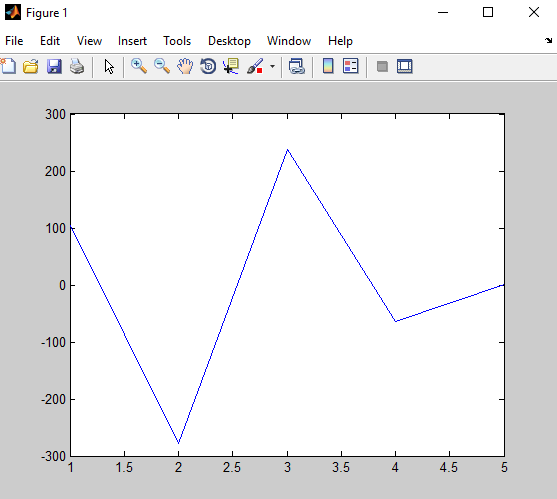
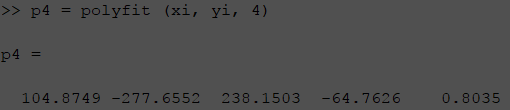
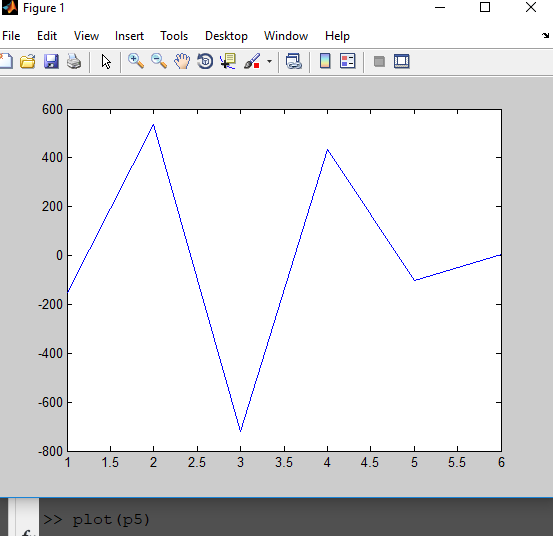
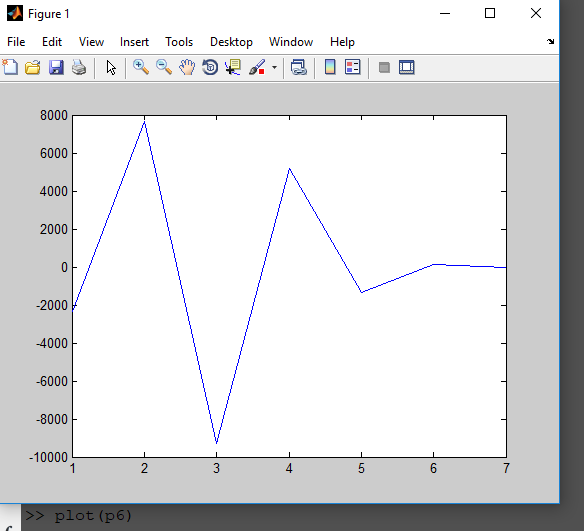
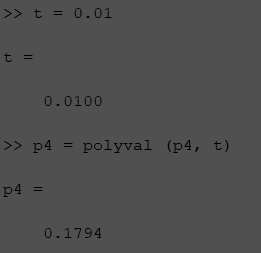


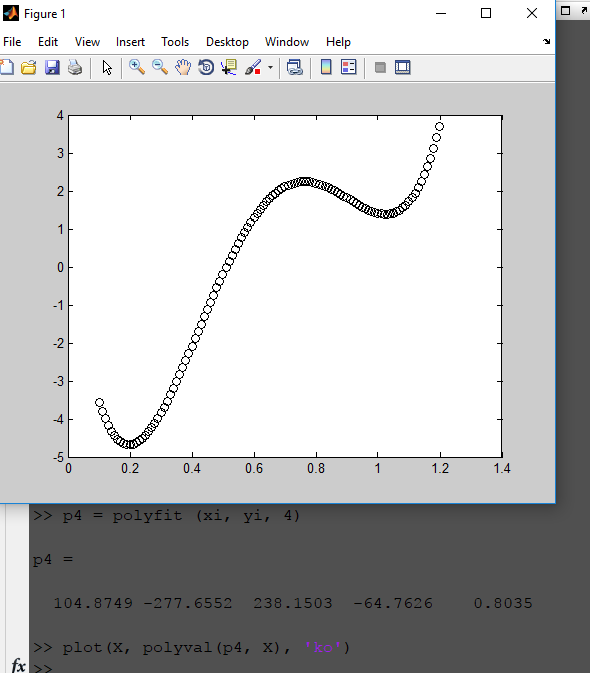
*Задание 8*. Исследуемая функция задана в виде таблицы:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| *xi* | 0,1 | 0,2 | 0,4 | 0,5 | 0,6 | 0,8 | 1,2 |
| *yi* | –3,5 | –4,8 | –2,1 | 0,2 | 0,9 | 2,3 | 3,7 |

Составьте файл-программу для приближения функции полиномами четвертой, пятой и шестой степени и выведите графики разными линиями, отражающими характер приближений.

*Методика выполнения*:

1. Ввести векторы *x* и *y*.
2. Вывести график табличной функции маркерами: *plot* (*x*, *y*, ‘*ko*’).
3. 
4. Вычислить коэффициенты полиномов разных степеней, приближающих табличную функцию, например *p*4 = *polyfit* (*x*, *y*, 4).
5. 
6. 
7. 
8. 
9. Задать шаг *t* =0,01.
10. Вычислить значение функций командой *polyval,* например *p*4 = *polyval* (*p*4*, t*).
11. 
12. Построить графики полиномов.
13. Сделать соответствующие пояснения к графикам: название, обозначение линий и т.д.

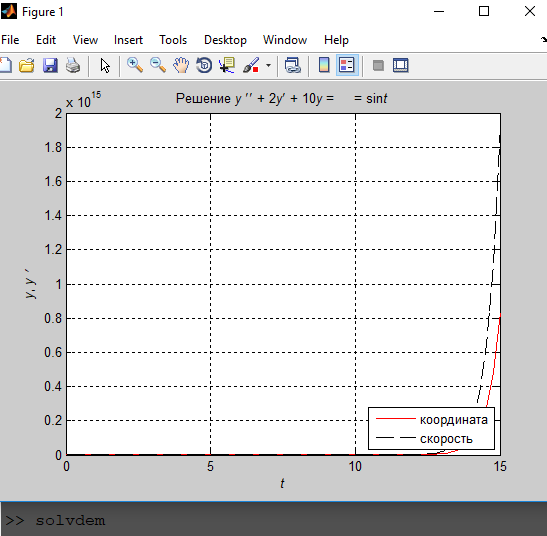


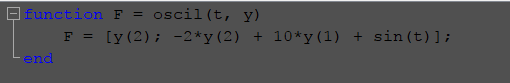
*Задание 9.* Решить задачу о колебаниях под воздействием внешней силы в среде, оказывающей сопротивление колебаниям. Уравнение, описывающее движение, имеет вид:



Координата точки в начальный момент времени равна 1, а скорость – нулю:

*y* (0) = 1, *y*′ (0) = 0.

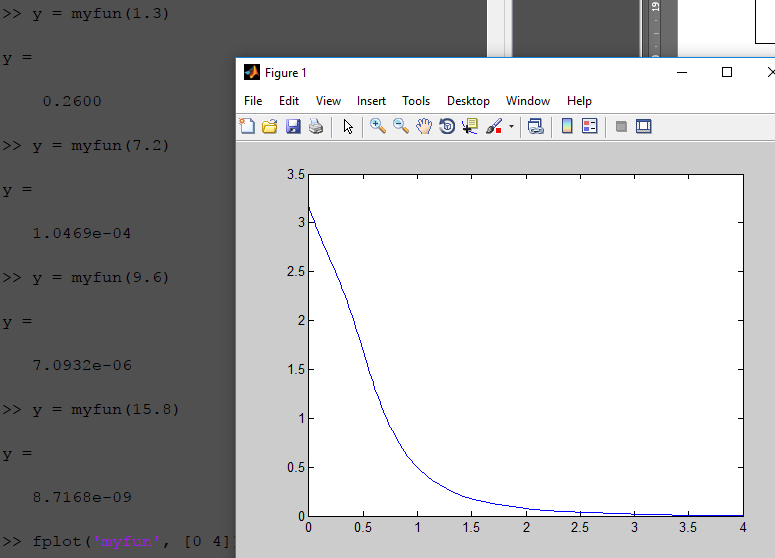




Самостоятельная работа № 1

Вычислите значение функции задания 3 в точках 1,3, 7,2, 9,6, 15,8. Постройте график функции *myfun* на отрезке [0, 4] при помощи файл-программы. Команда построения графика *fplot* (‘*myfun*’, [0 4]). График можно построить из командной строки или при помощи файл-программы. Кстати, *fplot* сама выбирает шаг аргумента.

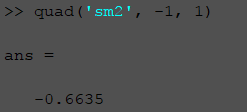
|  |  |
| --- | --- |
| Вариант № | Функция |
| 1 |  |



Самостоятельная работа № 2

Вычислить значение интеграла:

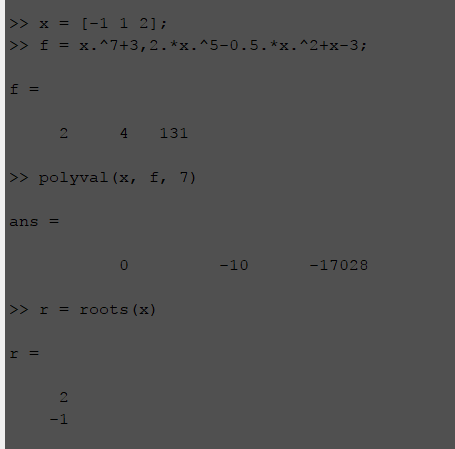
|  |  |
| --- | --- |
| Вариант № | Интеграл |
| 1 |  |



Самостоятельная работа № 3

Найти корни полинома и значения полинома в точках 1, –1, 2:

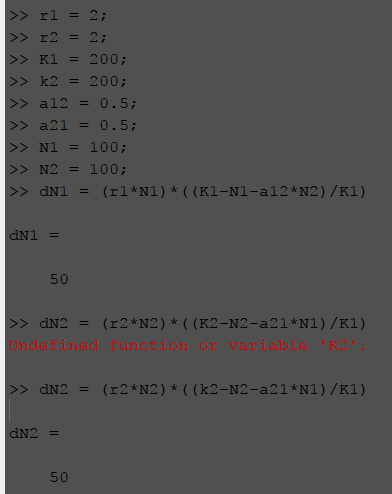
|  |  |
| --- | --- |
| Вариант № | Полином |
| 1 | *p* = *x*7 + 3,2*x*5 – 0,5 *x*2 + *x* – 3 |



Самостоятельная работа № 4

Вариант 1. Реализовать моделирование межвидовой конкуренции по формулам (1) при значениях параметров: *r*1 =2, *r*2 =2, *К*1 =200, *К*2 =200, α12 =0,5 и α12=0,5. Проанализировать зависимость судьбы популяций от соотношения значений их начальной численности *N*1 =100, *N*2 =100.





Увеличим первую популяцию:

>> r1 = 2;

>> r2 = 2;

>> K1 = 200;

>> K2 = 200;

>> a12 = 0.5;

>> a21 = 0.5;

>> N1 = 150;

>> N2 = 100;

>> dN1 = (r1\*N1)\*((K1-N1-a12\*N2)/K1)

dN2 = (r2\*N2)\*((K2-N2-a21\*N1)/K1)

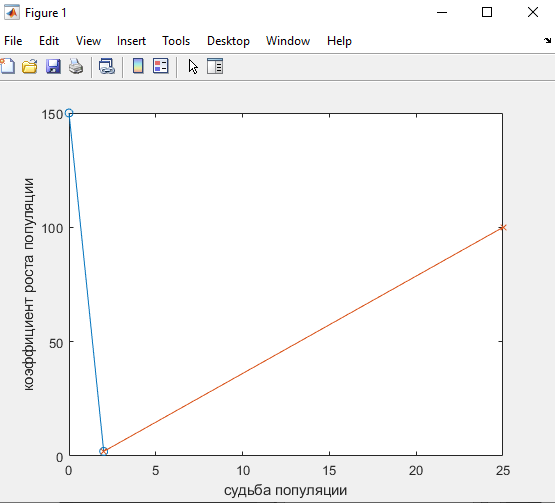
>> plot(N1, r1)

>> plot(N2, r2)

>> plot(dN1)

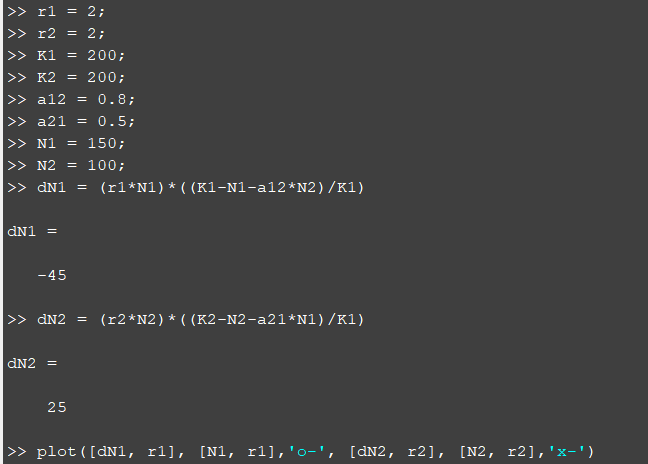
>> plot(dN1, dN2)

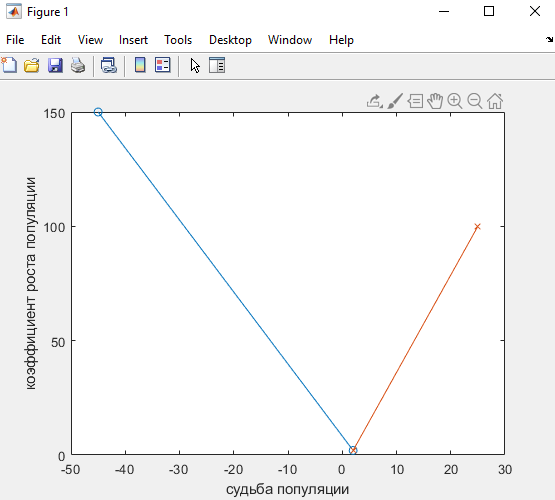
>> plot(dN2)



Вывод при увеличении начальной численности первой популяцию, первая популяция начинает вымирать, а вторая увеличивает свою популяцию.

Увеличим так же начальную популяцию один и увеличим коэффициента межвидовой конкуренции два:





Вывод: при увеличении начальной численности первой популяции и увеличении коэффициента роста первой популяции, первая популяция вымирает, а вторая увеличивает свое популяцию.

3.10. Контрольные вопросы

1. Что такое М-файл?

файлы скриптов

1. Как и для чего устанавливаются пути доступа к файлам?

Если не задать путь при доступе к файлу, первые поисковые запросы будут в текущей папке. Текущий каталог определяется в диалоговом окне Current Directory рабочей среды. Окно присутствует в рабочей среде, если выбран пункт Current Directory меню View рабочей среды. Текущий каталог выбирается из списка. Если его нет в списке, то его можно добавить из диалогового окна Browse for Folder, вызываемого нажатием на кнопку, расположенную справа от списка. Путь поиска определяется из диалогового окна командой меню File → Set Path.

1. Назначение файл-функции.

Это некоторые, скрипты (функции), которые пишутся в основном в отдельном файле и могут многократно вызываться для упрощения процесса вычисления.

1. Команды для создания файл-функции с одним или несколькими входными переменными.

С одним: *function* *r* = name(*x*)

С несколькими: *function* *r* = name (*x*, *y*, *z*)

1. Алгоритм создания файл-функции с несколькими выходными аргументами.

Раскройте меню File основного окна MatLab и в пункте New выберите подпункт M-file. Новый файл открывается в окне редактора М-файлов.

*function* *r* = name (*x*, *y*, *z*)

1. Как задается полином в MatLab?

Задается вектором его коэффициентов.

Нахождение значения полинома от некоторого аргумента осуществляется командой *polyval* (*p*, *xi*), где *p* – вектор коэффициентов, *xi* – аргумент.

1. Команда поиска локальных минимумов.

Поиск локальных минимумов осуществляется командой: [*xi f*] = *fminbnd* (‘*имя\_файл-функции*’, *отрезок*).

1. Алгоритм вычисления определенного интеграла.

При вычислении определенных интегралов первым шагом является создание файл-функции, вычисляющей подынтегральное выражение.

Команда вычисления интеграла: *I* = *quad* (‘*имя\_файл-функ-ции*’, *нижний\_предел*, *верхний\_предел*, *а*). Здесь *а* – число не равное нулю, задается для выбора узлов интегрирования.

Реализованный в команде алгоритм основан на квадратурной формуле Симпсона с автоматическим подбором шага интегрирования для достижения требуемой относительной погрешности.

1. Что такое интерполирование?

Алгоритм интерполяции определяется способом вычисления приближенных значений между точными. Наиболее простым и очевидным вариантом является построение прямой между двумя узловыми точками. Этот метод называется методом линейной интерполяции.

1. Алгоритм интерполирования в MatLab.

Ввести векторы *x* и *y*.

Вывести график табличной функции маркерами: *plot* (*x*, *y*, ‘*ko*’).

Вычислить коэффициенты полиномов разных степеней, приближающих табличную функцию, например *p*4 = *polyfit* (*x*, *y*, 4).

Задать шаг *t* =0,01.

Вычислить значение функций командой *polyval,* например *p*4 = *polyval* (*p*4*, t*).

Построить графики полиномов.

В Matlab такой способ реализован с помощью команды  
interp1(x,y, xi, 'linear') или просто interp1(x,y, xi), где **x** и **y** массивы из табличных данных (координаты точек), **xi** — массив промежуточных точек, координаты которых требуется найти.

1. Алгоритм решения дифференциальных уравнений.

Приведение дифференциального уравнения к системе дифференциальных уравнений первого порядка.

Написание специальной файл-функции для системы уравнений.

Вызов подходящего солвера.

Визуализация результата.