

# Отчёт по лабораторной работе 3

Елизавета Александровна Гайдамака

# Содержание

Цель работы	3
Задание	4
Теоретическое введение	5
Выполнение лабораторной работы	6
Выводы	15

## Цель работы

Целью данной работы является введение в работу с Octave.

# Задание

- Простейшие операции
- Операции с векторами
- Вычисление проектора
- Матричные операции
- Построение простейших графиков
- Два графика на одном чертеже
- График  $y = x^2 \sin x$
- Сравнение циклов и операций с векторами

# Теоретическое введение

GNU Octave — свободная программная система для математических вычислений, использующая совместимый с MATLAB язык высокого уровня.

Предоставляет интерактивный командный интерфейс для решения линейных и нелинейных математических задач, а также проведения других численных экспериментов. Кроме того, Octave можно использовать для пакетной обработки. Язык Octave оперирует арифметикой вещественных и комплексных скаляров, векторов и матриц, имеет расширения для решения линейных алгебраических задач, нахождения корней систем нелинейных алгебраических уравнений, работы с полиномами, решения различных дифференциальных уравнений, интегрирования систем дифференциальных и дифференциально-алгебраических уравнений первого порядка, интегрирования функций на конечных и бесконечных интервалах. Этот список можно легко расширить, используя язык Octave (или используя динамически загружаемые модули, созданные на Си, C++, Фортране и других).

Система написана на C++ с использованием стандартной библиотеки шаблонов. Для запуска сценариев используется встроенный интерпретатор. Для создания и отображения графиков можно использовать на выбор gnuplot, встроенный модуль на основе FLTK, или Grace[en].

Название дано в честь профессора химического машиностроения Октава Левеншпиля.

# Выполнение лабораторной работы

1. Включите журналирование сессии.
2. Octave можно использовать как простейший калькулятор. Вычислим выражение

$$2 \times 6 + (7 - 4)^2$$

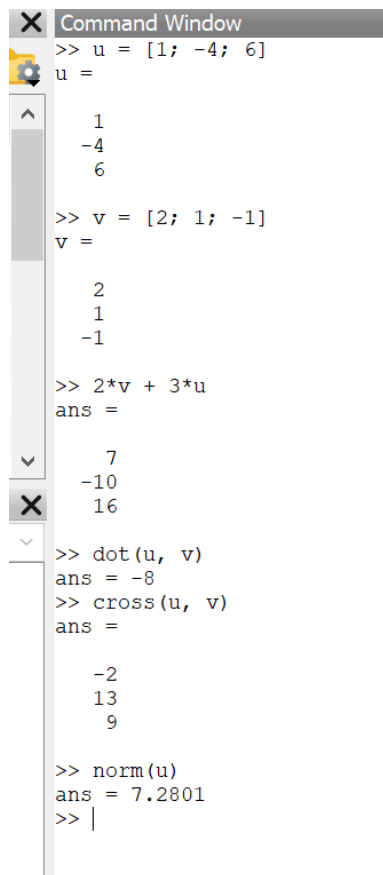
3. Зададим вектор-строку (ковектор).
4. Аналогично можно задать вектор-столбец (вектор).
5. Теперь зададим матрицу.

```
>> diary on
>> 2*6 + (7-4)^2
ans = 21
>> u = [1 -4 6]
u =
    1   -4    6
>> u = [1; -4; 6]
u =
     1
    -4
     6
>> A = [1 2 -3; 2 4 0; 1 1 1]
A =
     1     2    -3
     2     4     0
     1     1     1
>> |
```

Рис. 1: Рис.1

1. Зададим два вектора-столбца

2. Сложение векторов
3. Скалярное умножение векторов
4. Векторное умножение векторов
5. Норма вектора



```
Command Window
>> u = [1; -4; 6]
u =
     1
    -4
     6

>> v = [2; 1; -1]
v =
     2
     1
    -1

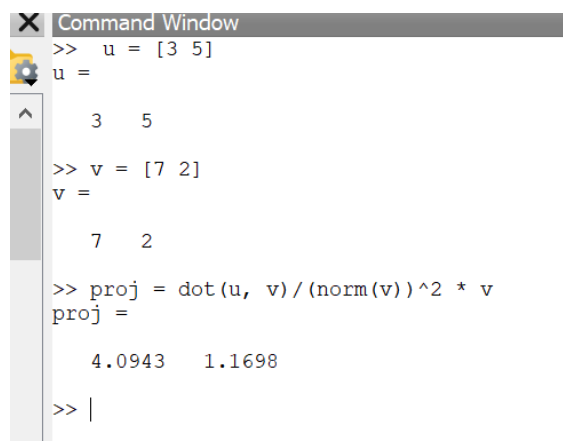
>> 2*v + 3*u
ans =
     7
    -10
    16

>> dot(u, v)
ans = -8
>> cross(u, v)
ans =
    -2
    13
     9

>> norm(u)
ans = 7.2801
>> |
```

Рис. 2: Рис.2

1. Введём два вектора-строки
2. Вычислим проекцию вектора на вектор

A screenshot of the MATLAB Command Window. The window title is 'Command Window'. The command history shows:   
1. `>> u = [3 5]` followed by `u =` and the output `3 5`.   
2. `>> v = [7 2]` followed by `v =` and the output `7 2`.   
3. `>> proj = dot(u, v) / (norm(v))^2 * v` followed by `proj =` and the output `4.0943 1.1698`.   
4. The prompt `>> |` is shown at the bottom.

```
X Command Window
>> u = [3 5]
u =
    3    5
>> v = [7 2]
v =
    7    2
>> proj = dot(u, v) / (norm(v))^2 * v
proj =
    4.0943    1.1698
>> |
```

Рис. 3: Рис.3

1. Введём матрицы A и B
2. Вычислим произведение матриц A и B
3. Вычислим произведение матриц A и B
4. Вычислим  $2A - 4I$
5. Найдём определитель матрицы A
6. Найдём обратную матрице A
7. Найдём собственные значения матрицы A
8. Найдём ранг матрицы A



```
Command Window
>> A = [1 2 -3; 2 4 0; 1 1 1]
A =
    1     2    -3
    2     4     0
    1     1     1

>> B = [1 2 3 4; 0 -2 -4 6; 1 -1 0 0]
B =
    1     2     3     4
    0    -2    -4     6
    1    -1     0     0

>> A * B
ans =
   -2     1    -5    16
     2    -4   -10    32
     2    -1    -1     10

>> B' * A
ans =
     2     3    -2
    -3    -5    -7
    -5   -10    -9
    16    32   -12

>> 2 * A - 4 * eye(3)
ans =
   -2     4    -6
     4     4     0
     2     2    -2

>> eye(3)
ans =
     1     0     0
     0     1     0
     0     0     1

>> det(A)
ans = 6
```

Рис. 4: Рис.4

```

>> inv (A)
ans =

    0.6667    -0.8333    2.0000
   -0.3333     0.6667   -1.0000
   -0.3333     0.1667     0.0000

>> eig (A)
ans =

    4.5251 + 0i
    0.7374 + 0.8844i
    0.7374 - 0.8844i

>> rank (A)
ans = 3
>> |

```

Рис. 5: Рис.5

1. Создадим вектор значений  $x$
2. Зададим вектор  $y = \sin x$
3. Построим график
4. Улучшим внешний вид графика
5. Зададим красный цвет для линии и сделаем её потолще
6. Подгоним диапазон осей
7. Нарисуем сетку
8. Подпишем оси
9. Сделаем заголовок графика
10. Зададим легенду

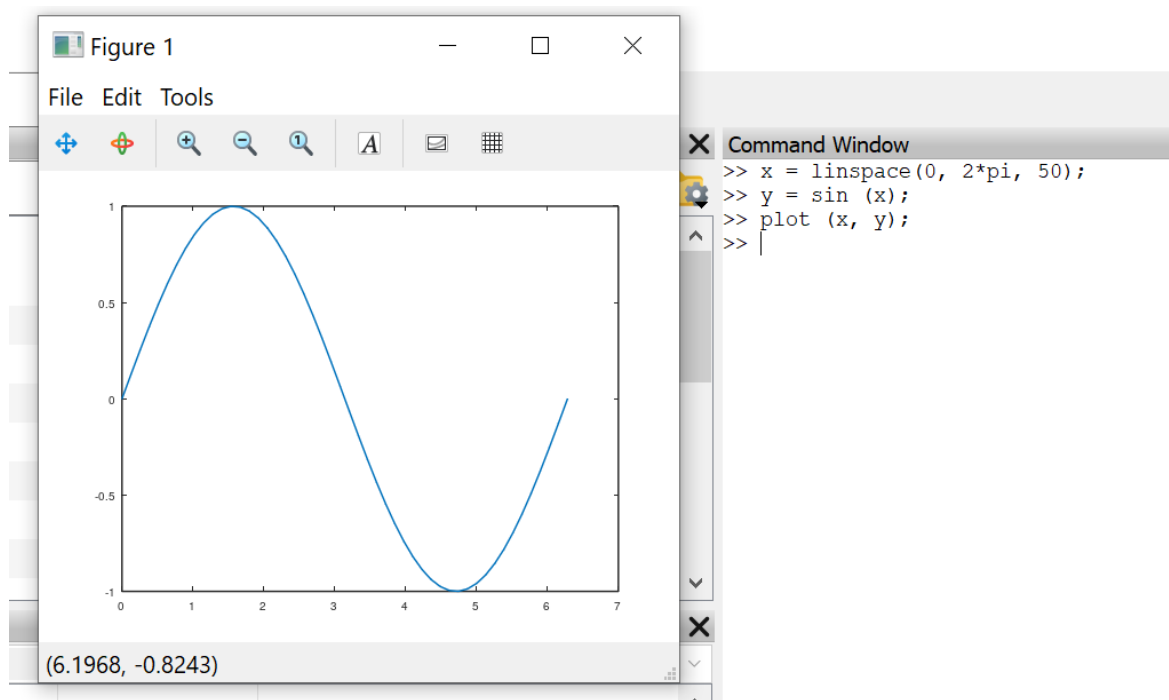


Рис. 6: Рис.6

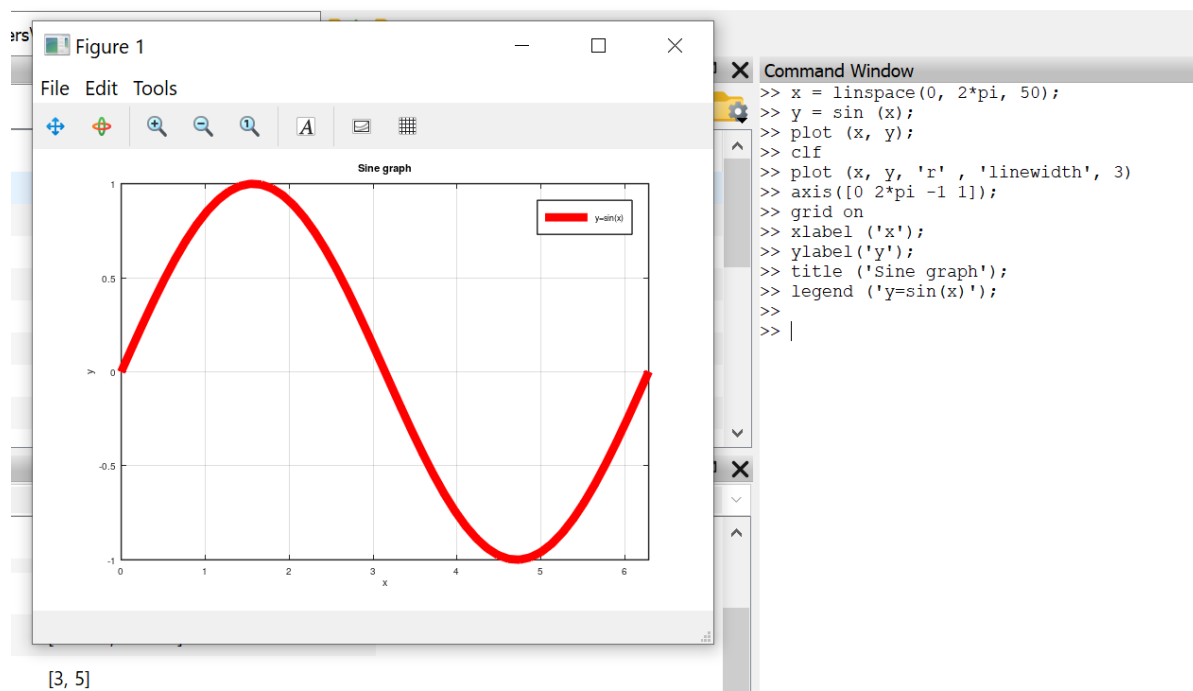


Рис. 7: Рис.7

1. Очистим память и рабочую область фигуры
2. Зададим два вектора
3. Начертим эти точки, используя кружочки как маркеры
4. Чтобы добавить к нашему текущему графику ещё один, нужно использовать команду `hold on`
5. Добавим график регрессии
6. Зададим сетку, оси и легенду

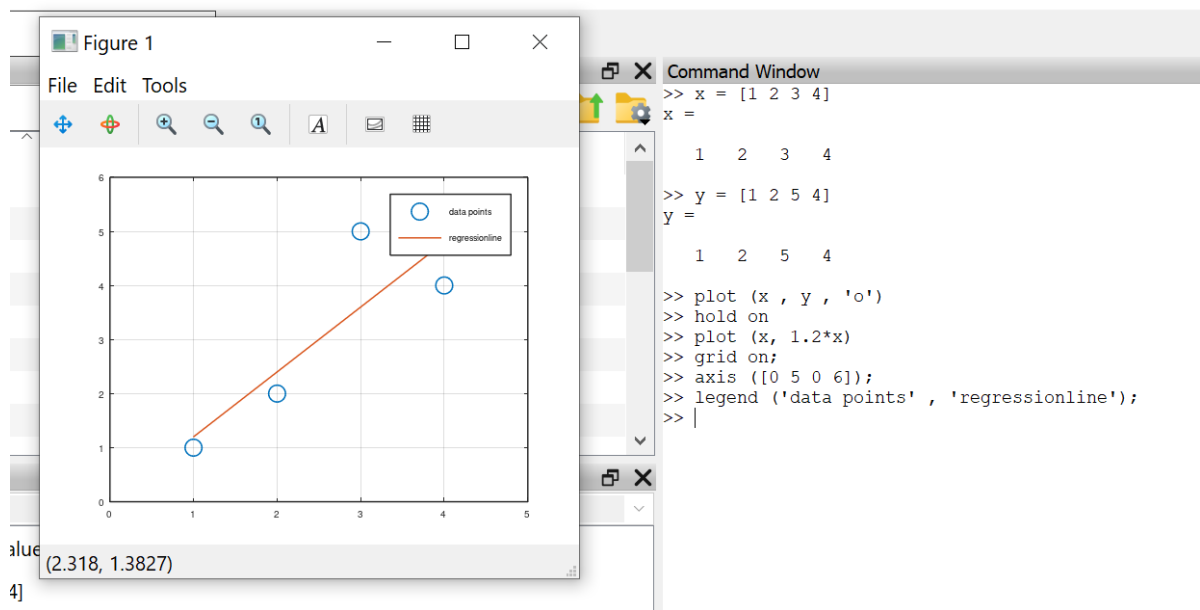


Рис. 8: Рис.8

1. Очистим память и рабочую область фигуры
2. Зададим вектор
3. Построим график  $y = x^2 \cdot \sin x$ , используя поэлементное возведение в степень и поэлементное умножение
4. Сохраним графики в виде файлов

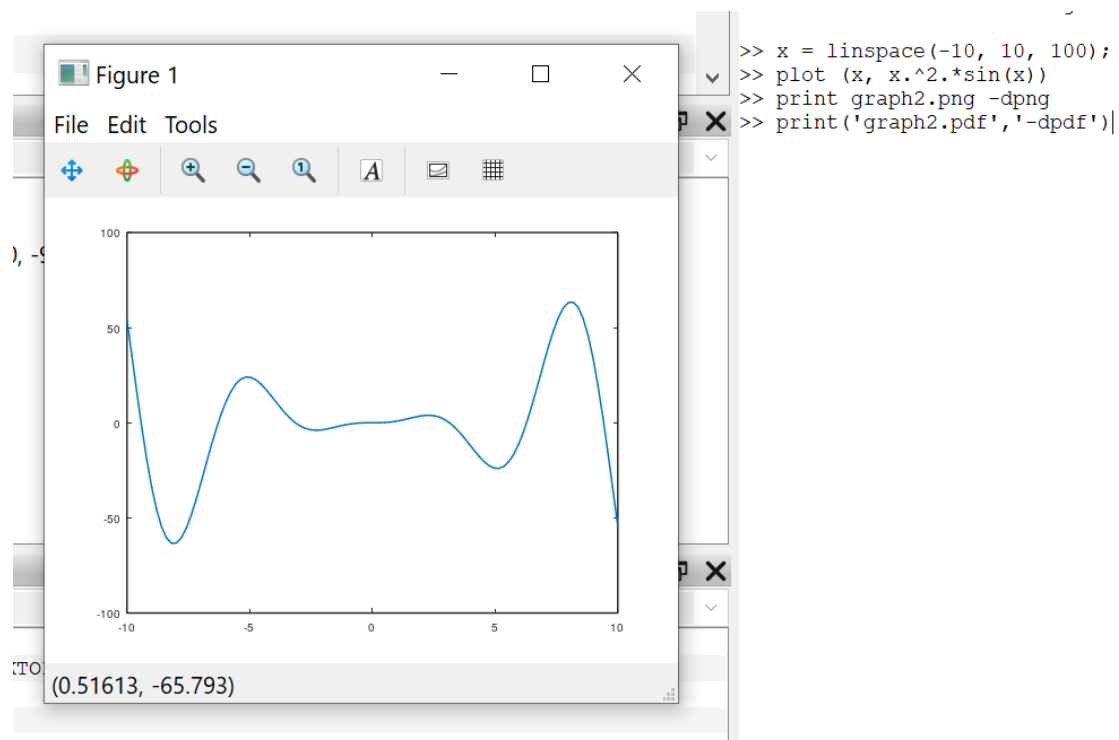


Рис. 9: Рис.9

1. Очистим память и рабочую область фигуры
2. Вычислим сумму (3.1) с помощью цикла. Создадим файл loop\_for.m
3. Запустим файл loop\_for.m. В окне команд получим вывод
4. Вычислим сумму (3.1) с помощью операций с векторами. Создадим файл loop\_vec.m
5. Запустим файл loop\_vec.m. В окне команд получим вывод Elapsed time is 0.040108 seconds.

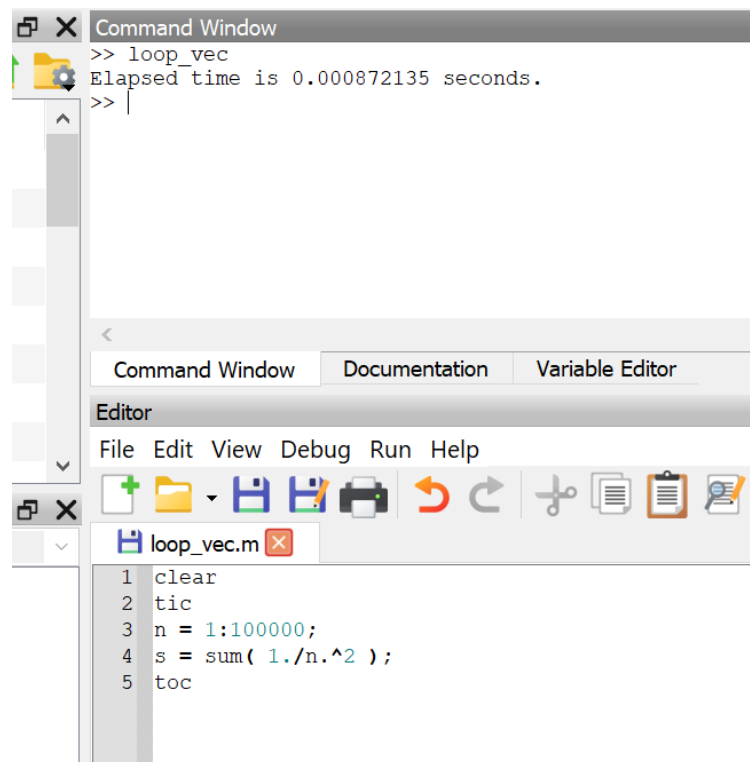


Рис. 10: Рис.10

## 6. Завершим запись в файл

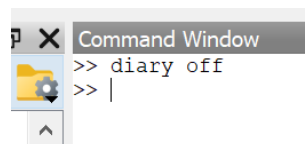


Рис. 11: Рис.11

## Выводы

Благодаря данной работе я ознакомилась с основными командами Octave.