Лабораторная работа 3. Введение в работу с Octave

Отчет по лабораторной работе 3

Хитяев Евгений Анатольевич НПМмд-02-21

Содержание

[1 Цель работы 1](#_Toc88863564)

[2 Теоретические сведения 1](#_Toc88863565)

[3 Задание 1](#_Toc88863566)

[4 Выполнение лабораторной работы 2](#_Toc88863567)

[5 Выводы 22](#_Toc88863568)

# 1 Цель работы

Познакомиться с интерфейсом Octave.

# 2 Теоретические сведения

Octave является свободной реализацией языка MATLAB. Графический интерфейс Octave похож на графический интерфейс MATLAB. Язык MATLAB был разработан Кливом Моулером (англ. Cleve Moler) в конце 1970-х годов. Целью разработки служила задача дать студентам факультета возможность использования программных библиотек Linpack и EISPACK без необходимости изучения языка FORTRAN. Язык распространился среди других университетов и был с большим интересом встречен учёными, работающими в области прикладной математики. MATLAB широко используется для выполнения инженерных и научных расчётов, а также в образовании. В 1984 году была основана компания The MathWorks для коммерциализации MATLAB.

Вся теоритическая часть по использованию интерфейса Octave была взята из инструкции по лабораторной работе №3 на сайте: https://esystem.rudn.ru/pluginfile.php/1284124/mod\_resource/content/4/003-octave-intro.pdf

# 3 Задание

Выполните работу и задокументируйте процесс выполнения.

# 4 Выполнение лабораторной работы

**1. Простейшие операции**

* Включим журналирование сессии (см. рис. 1).

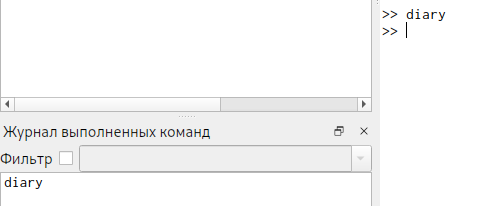


Figure 1: Журналирование сессии

* Продемонстрируем, что Octave можно использовать как простейший калькулятор. Для этого вычислим выражение (см. рис. 2).

Figure 2: Вычисление выражения

Figure 2: Вычисление выражения

* Зададим вектор-строку (ковектор) (см. рис. 3).

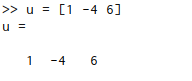


Figure 3: Задание вектора-строки (ковектора)

* Зададим вектор-столбец (вектор) (см. рис. 4).

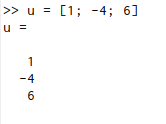


Figure 4: Задание вектора-столбца (вектора)

* Зададим матрицу (см. рис. 5).

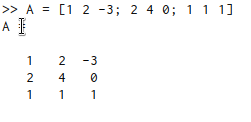


Figure 5: Задание матрицы

**2. Операции с векторами**

* Зададим два вектора-столбца (см. рис. 6).

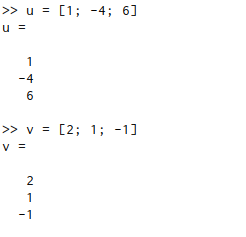


Figure 6: Задание двух векторов-столбцов

* Выполним операцию сложения векторов (см. рис. 7).

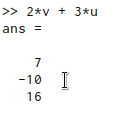


Figure 7: Выполнение операции сложения векторов

* Произведем скалярное умножение векторов (см. рис. 8).

Figure 8: Скалярное умножение векторов

Figure 8: Скалярное умножение векторов

* Произведем векторное умножение (см. рис. 9).

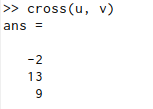


Figure 9: Векторное умножение

* Вычислим норму вектора (см. рис. 10).

Figure 10: Вычисление нормы вектора

Figure 10: Вычисление нормы вектора

**3. Вычисление проектора**

* Введем два вектора-строки (см. рис. 11).

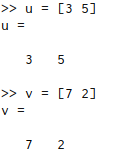


Figure 11: Задание двух векторов-строк

* Вычислим проекцию вектора u на вектор v (см. рис. 12).

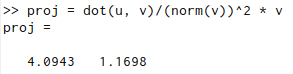


Figure 12: Вычисление проекции вектора u на вектор v

**4. Матричные операции**

* Введем матрицы Â и B (см. рис. 13).

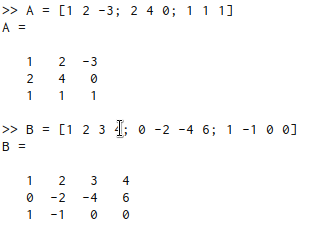


Figure 13: Введение двух матриц Â и B

* Вычислим произведение матриц ÂB (см. рис. 14).

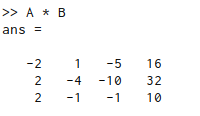


Figure 14: Вычисление произведения матриц ÂB

* Вычислим произведение матриц B̂TÂ.(см. рис. 15).

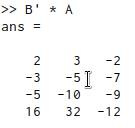


Figure 15: Вычисление произведения матриц B̂TÂ

* Вычислим 2Â - 4Î, где Î есть единичная матрица (см. рис. 16).

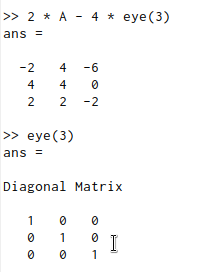


Figure 16: Вычисление выражения

* Найдем определитель |Â| (см. рис. 17).

Figure 17: Нахождение определителя

Figure 17: Нахождение определителя

* Найдем обратную матрицу Â−1 (см. рис. 18).

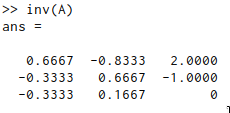


Figure 18: Нахождение обратной матрицы

* Найдем собственные значения матрицы (см. рис. 19).

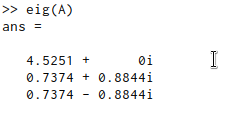


Figure 19: Нахождение собственных значений матрицы

* Вычислим ранг матрицы (см. рис. 20).

Figure 20: Вычисление ранга матрицы

Figure 20: Вычисление ранга матрицы

**5. Построение простейших графиков**

* Построим график функции sin(x) на интервале [0, 2π]. Создадим вектор значений x (см. рис. 21).

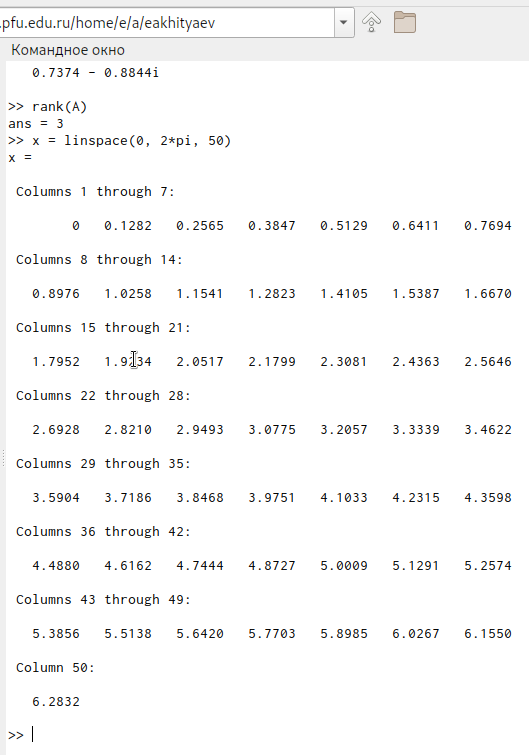


Figure 21: Создание вектора значений x

* Зададим вектор y = sin(x) (см. рис. 22).

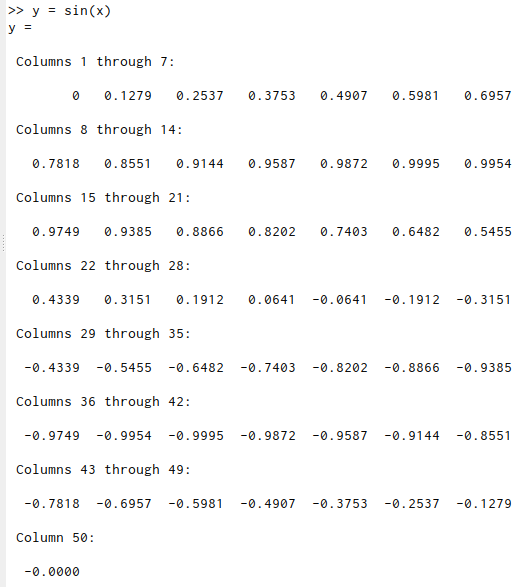


Figure 22: Задание вектора y = sin(x)

* Построим график (см. рис. 23.1, 23.2).

Figure 23: Построение графика y = sin(x)

Figure 23: Построение графика y = sin(x)

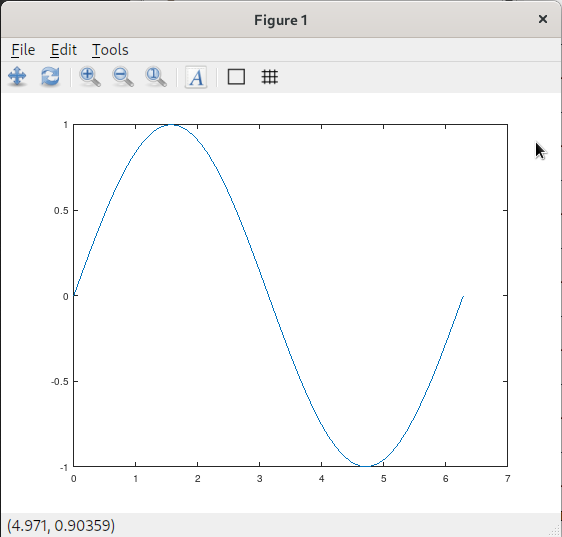


Figure 24: График y = sin(x)

* Улучшим внешний вид графика. Сначала очистим получившийся график (см. рис. 24.1). Заметим, что заданные вектора x и y сохранились (см. рис. 24.2).

Figure 25: Очистка графика

Figure 25: Очистка графика

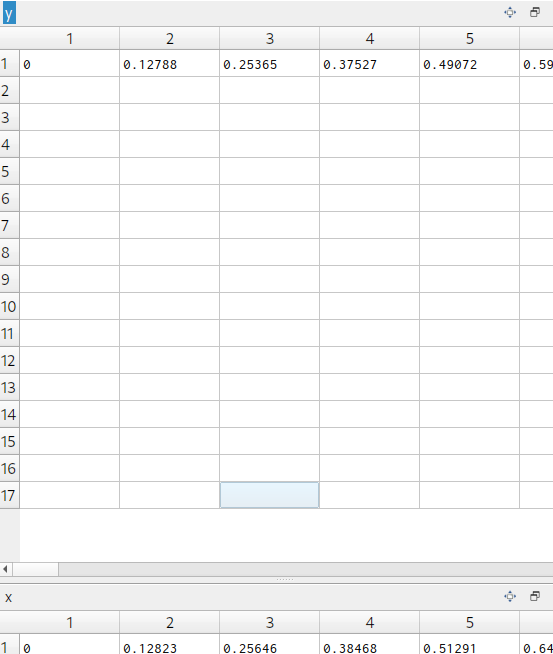


Figure 26: Вектора x и y

* Зададим красный цвет для линии и сделаем ее потолще (см. рис. 25.1, 25.2).

Figure 27: Задание цвета и размера линии

Figure 27: Задание цвета и размера линии

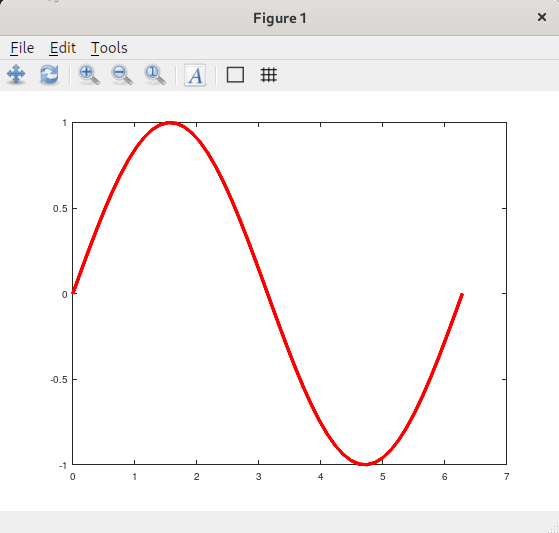


Figure 28: График y = sin(x) после изменения цвета и размера линии

* Подгоним диапазон осей (см. рис. 26.1, 26.2).

Figure 29: Подгонка диапазона осей

Figure 29: Подгонка диапазона осей

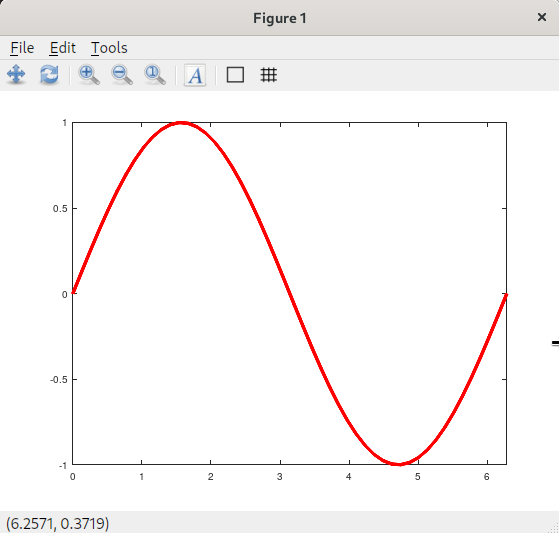


Figure 30: График y = sin(x) после подгонки осей

* Нарисуем сетку (см. рис. 27.1, 27.2).

Figure 31: Отрисовка сетки

Figure 31: Отрисовка сетки

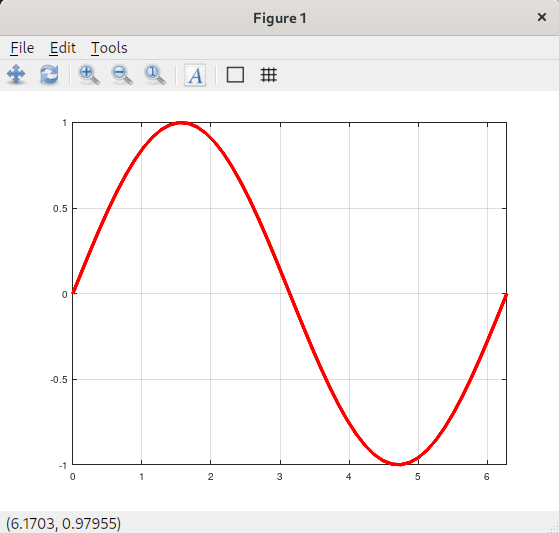


Figure 32: График y = sin(x) после отрисовки сетки

* Подпишем оси (см. рис. 28.1, 28.2).

Figure 33: Подпись осей

Figure 33: Подпись осей

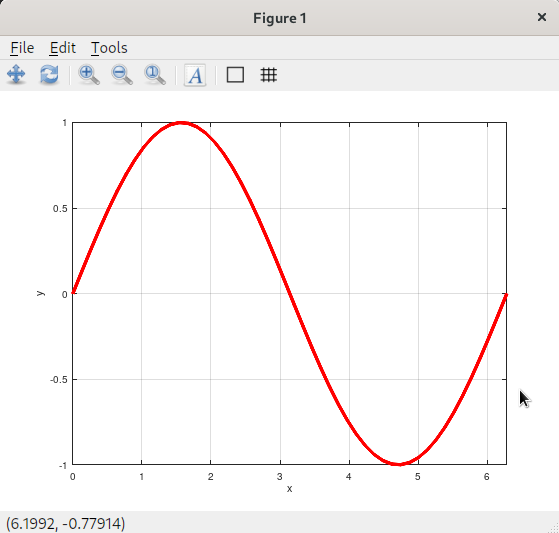


Figure 34: График y = sin(x) после подписи осей

* Сделаем заголовок графика и зададим легенду (см. рис. 29). В результате получим следующий график (см. рис. 30).

Figure 35: Создание заголовка графика и задание легенды

Figure 35: Создание заголовка графика и задание легенды

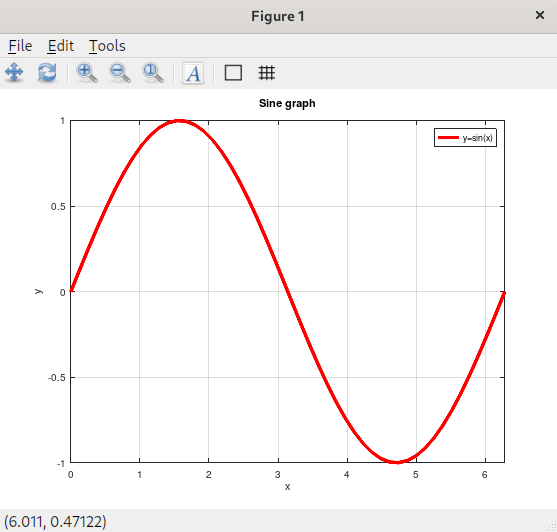


Figure 36: График y = sin(x) после создания заголовка и задания легенды

**6. Два графика на одном чертеже**

* Начертим два графика на одном чертеже. Очистим память и рабочую область фигуры (см. рис. 31).

Figure 37: Очистка памяти и рабочей области фигуры

Figure 37: Очистка памяти и рабочей области фигуры

* Зададим два вектора (см. рис. 32).

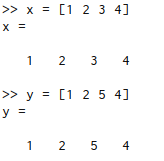


Figure 38: Задание двух векторов

* Начертим эти точки, используя кружочки, как маркеры (см. рис. 33, 34).

Figure 39: Чертеж точек

Figure 39: Чертеж точек

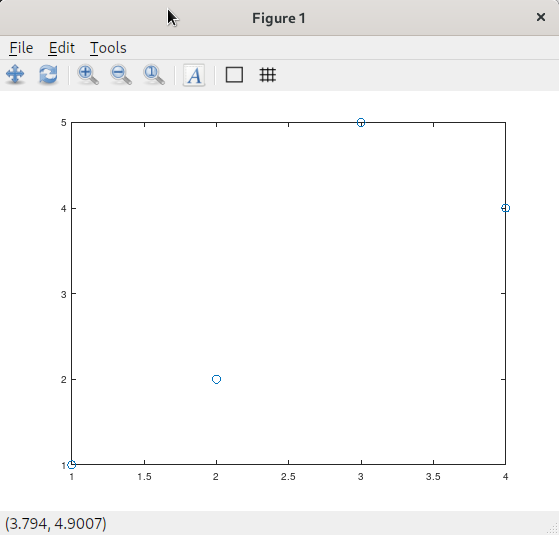


Figure 40: График с отрисованными точками

* Чтобы добавить к нашему текущему графику ещё один, нужно использовать команду hold on (см. рис. 35).

Figure 41: Использование команды hold on

Figure 41: Использование команды hold on

* Добавим график регрессии (см. рис. 36, 37).

Figure 42: Добавление дополнительного графика

Figure 42: Добавление дополнительного графика

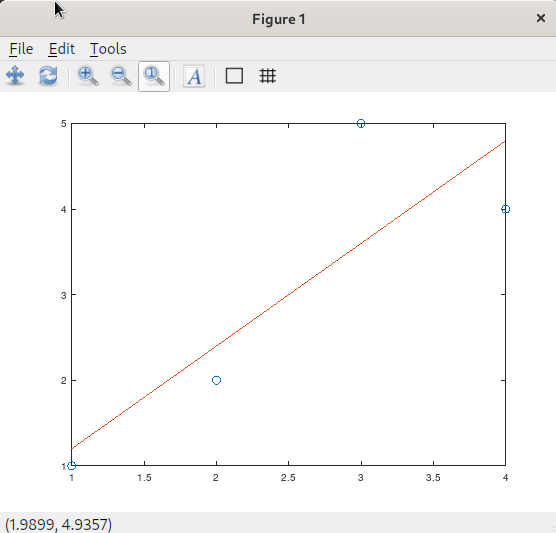


Figure 43: Исходный и добавленный графики

* Зададим сетку, оси и легенду (см. рис. 38). В результате получим следующий график (см. рис. 39).

Figure 44: Задание сетки, оси и легенды

Figure 44: Задание сетки, оси и легенды

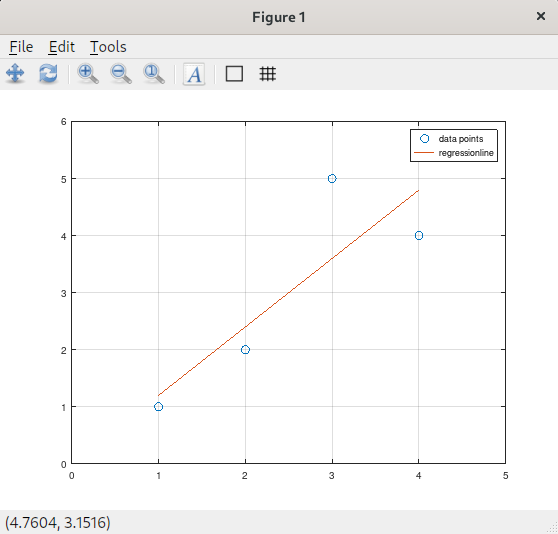


Figure 45: График после задания сетки, оси и легенды

**7. График y=x2sin(x)**

* Очистим память и рабочую область фигуры (см. рис. 40, 41).

Figure 46: Очистка памяти и рабочей области фигуры

Figure 46: Очистка памяти и рабочей области фигуры

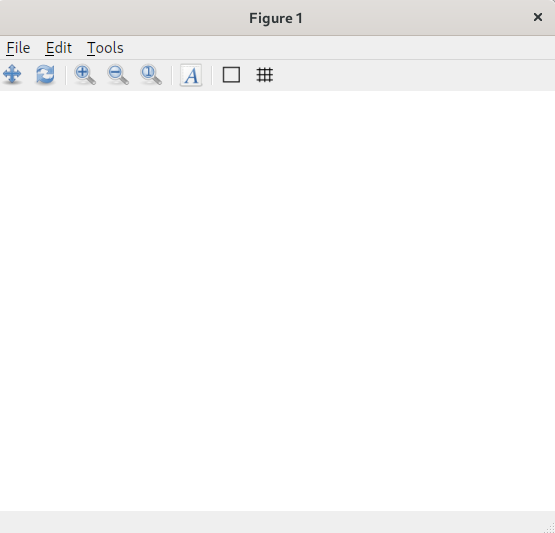


Figure 47: Очищенная область

* Зададим вектор x (см. рис. 42).

Figure 48: Задание вектора x

Figure 48: Задание вектора x

* Построим график y=x2sin(x) (см. рис. 43).

Figure 49: Построение графика y=x2sin(x)

Figure 49: Построение графика y=x2sin(x)

Ничего не получилось. Действительно, мы задали в выражении матричное умножение. В то время, как нам необходимо поэлементное.

* Построим график y=x2sin(x), используя поэлементное возведение в степень .^ и поэлементное умножение (см. рис. 44, 45).

Figure 50: Построение графика y=x2sin(x) с поэлементными возведением в степень и умножением

Figure 50: Построение графика y=x2sin(x) с поэлементными возведением в степень и умножением

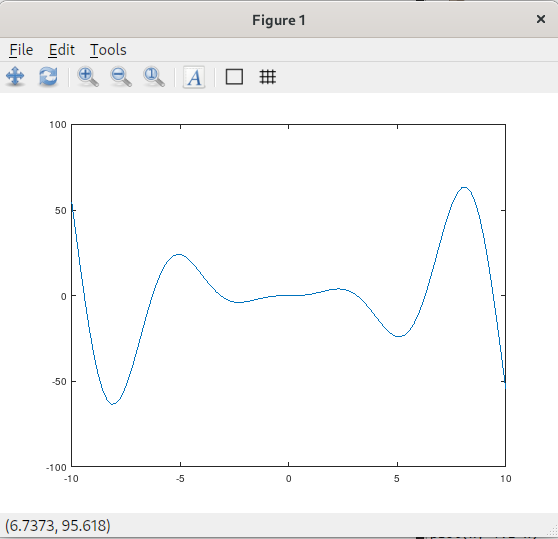


Figure 51: График после построения

* Сохраним графики в виде файлов (см. рис. 46).

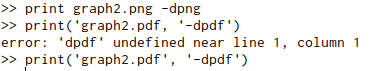


Figure 52: Сохранение графиков

**8. Сравнение циклов и операций с векторами**

* Сравним эффективность работы с циклами и операций с векторами. Для этого вычислим сумму 3.1 (см. рис. 47).

Figure 53: Сумма

Figure 53: Сумма

* Очистим память и рабочую область фигуры (см. рис. 48). Вычислим сумму с помощью цикла, создадим файл loop-for.m, функции tic и toc служат для запуска и остановки таймера (см. рис. 49).

Figure 54: Очистка памяти и рабочей области фигуры

Figure 54: Очистка памяти и рабочей области фигуры

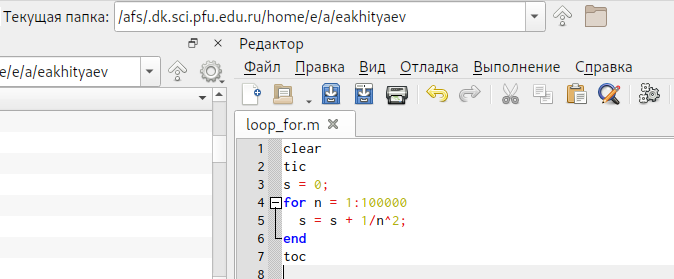


Figure 55: Создание файла loop\_for.m

* Запустим файл loop-for.m (см. рис. 50).

Figure 56: Запуск файла loop_for.m

Figure 56: Запуск файла loop\_for.m

* Вычислим сумму с помощью операций с векторами. Создадим файл loop-vec.m (см. рис. 51), запустим его (см. рис. 52).

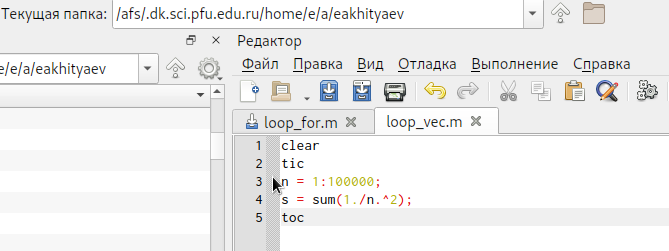


Figure 57: Создание файла loop\_vec.m

Figure 58: Запуск файла loop_vec.m

Figure 58: Запуск файла loop\_vec.m

Во втором случае сумма вычисляется значительно быстрее.

* Завершим запись в файл (см. рис. 53).

Figure 59: Завершение записи в файл

Figure 59: Завершение записи в файл

# 5 Выводы

В ходе выполнения работы я познакомился с некоторыми простейшими операциями в Octave.