

# **Лабораторная работа №3**

**Научное программирование**

Таубер Кирилл Олегович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Теоретическое введение</b>	<b>5</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>4</b>	<b>Вывод</b>	<b>17</b>
	<b>Список литературы</b>	<b>18</b>

## Список иллюстраций

3.1	Простейшие операции . . . . .	7
3.2	Операции с векторами . . . . .	8
3.3	Вычисление проекции одного вектора на другой . . . . .	8
3.4	Матричные операции . . . . .	9
3.5	Матричные операции . . . . .	10
3.6	Построение простейших графиков . . . . .	11
3.7	Построение простейших графиков . . . . .	12
3.8	Два графика на одном чертеже . . . . .	12
3.9	График $y = x^2 \sin(x)$ . . . . .	13
3.10	Вычисление суммы с помощью цикла . . . . .	14
3.11	Вычисление суммы с помощью операций с векторами . . . . .	15
3.12	Сравнение двух программ . . . . .	16

# 1 Цель работы

Освоить базовые навыки работы в Octave: простейшие вычислительные операции, операции с векторами и матрицами, построение простейших графиков, сравнение циклов и операций с векторами.

## 2 Теоретическое введение

**Octave** — высокоуровневый интерпретируемый язык программирования, предназначенный для решения задач вычислительной математики. По возможностям и качеству реализации интерпретатора язык Octave можно сравнивать с проприетарной программой MATLAB, причём синтаксис обоих языков очень схож. В состав пакета входит интерактивный командный интерфейс (интерпретатор Octave). Интерпретатор Octave запускается из терминала ОС Linux или из его порта в Windows. После запуска Octave пользователь видит окно интерпретатора.

В окне интерпретатора пользователь может вводить как отдельные команды языка Octave, так и группы команд, объединяемые в программы. Если строка заканчивается символом “;”, результаты на экран не выводятся. Если же в конце строки символ “;” отсутствует, результаты работы выводятся на экран. Текст в строке после символа % является комментарием и интерпретатором не обрабатывается.

Octave имеет большое количество инструментов для решения распространенных задач числовой линейной алгебры, поиска корней нелинейных уравнений, интегрирования обычных функций, управления полиномами, интегрирования обыкновенных дифференциальных и дифференциально-алгебраических уравнений и т.д. Помимо этого в Octave можно рисовать графики. Функционал этой программной системы может быть легко расширен и перенастроен с помощью пользовательских функций, написанных на собственном языке программирования Octave, либо с помощью динамически загружаемых модулей, написанных на C++, C, Fortran или других языках.

В Octave реализованы многие возможности Matlab, включая использование матриц в качестве основных типов данных, поддержку комплексных чисел, поддержку математических функции и больших библиотек функций, а также возможность создания пользовательских функций для расширения функциональности системы.

Более подробно см. в [[@Octave\\_1:bash](#)] и [[@Octave\\_2:bash](#)].

### 3 Выполнение лабораторной работы

Для начала включили журналирование сессии, создав документ “diary”. Вычислили значение выражения. Задали вектор-строку (ковектор), вектор-столбец (вектор) и матрицу (рис. fig. 3.1).

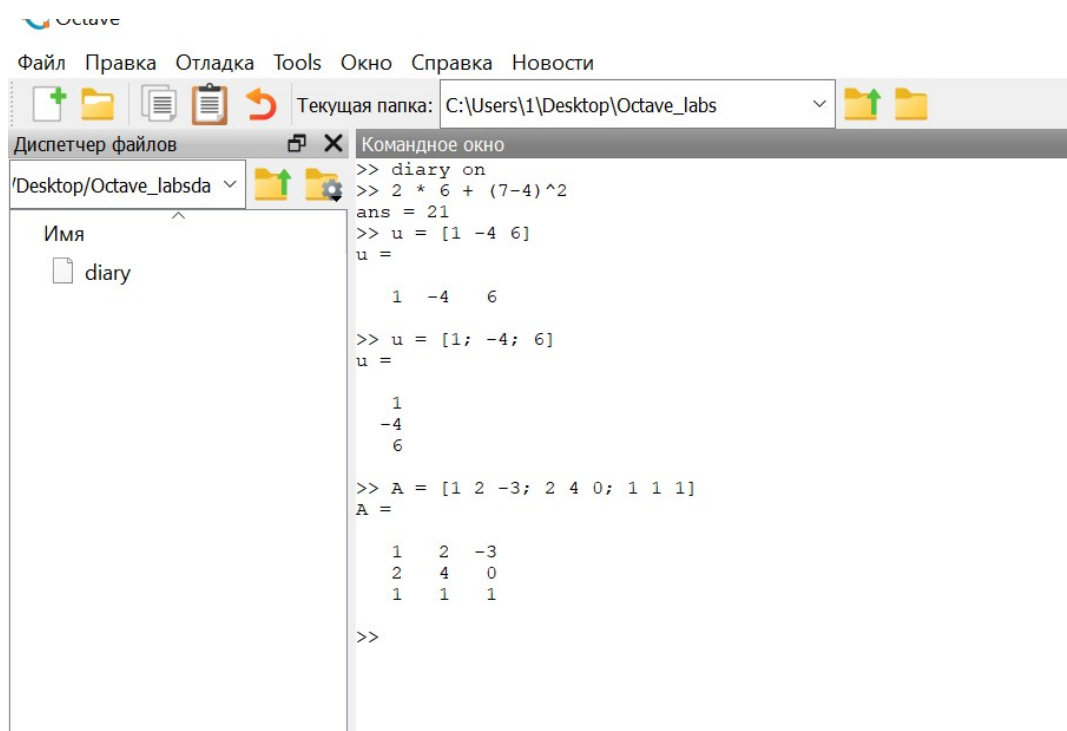


Рис. 3.1: Простейшие операции

Задали два вектора-столбца. Осуществили сложение заданных векторов, их скалярное и векторное умножение. Нашли норму одного из векторов (рис. fig. 3.2).

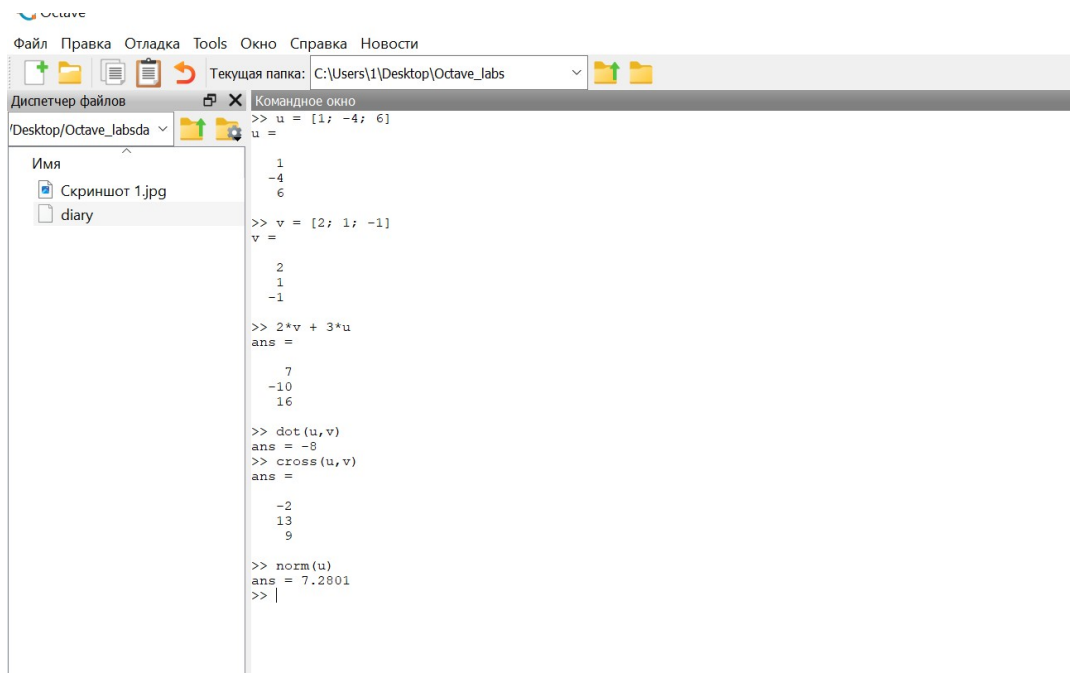


Рис. 3.2: Операции с векторами

Ввели два вектора-строки. Вычислили проекцию одного вектора на другой (рис. fig. 3.3).

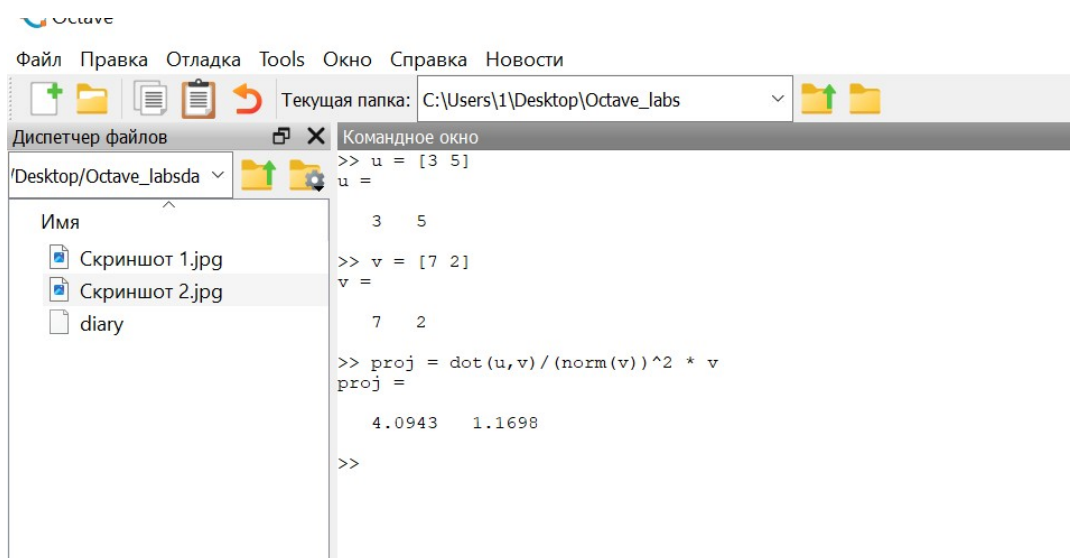


Рис. 3.3: Вычисление проекции одного вектора на другой

Ввели две матрицы. Вычислили их произведение, а также их произведение с учетом транспонирования одной из матриц. Вычислили выражение, содер-



жащее единичную матрицу. Нашли определитель матрицы, обратную для нее, вычислили собственные значения матрицы и ее ранг (рис. fig. 3.4 и рис. fig. 3.5).

```

>> 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
  
```

Рис. 3.4: Матричные операции

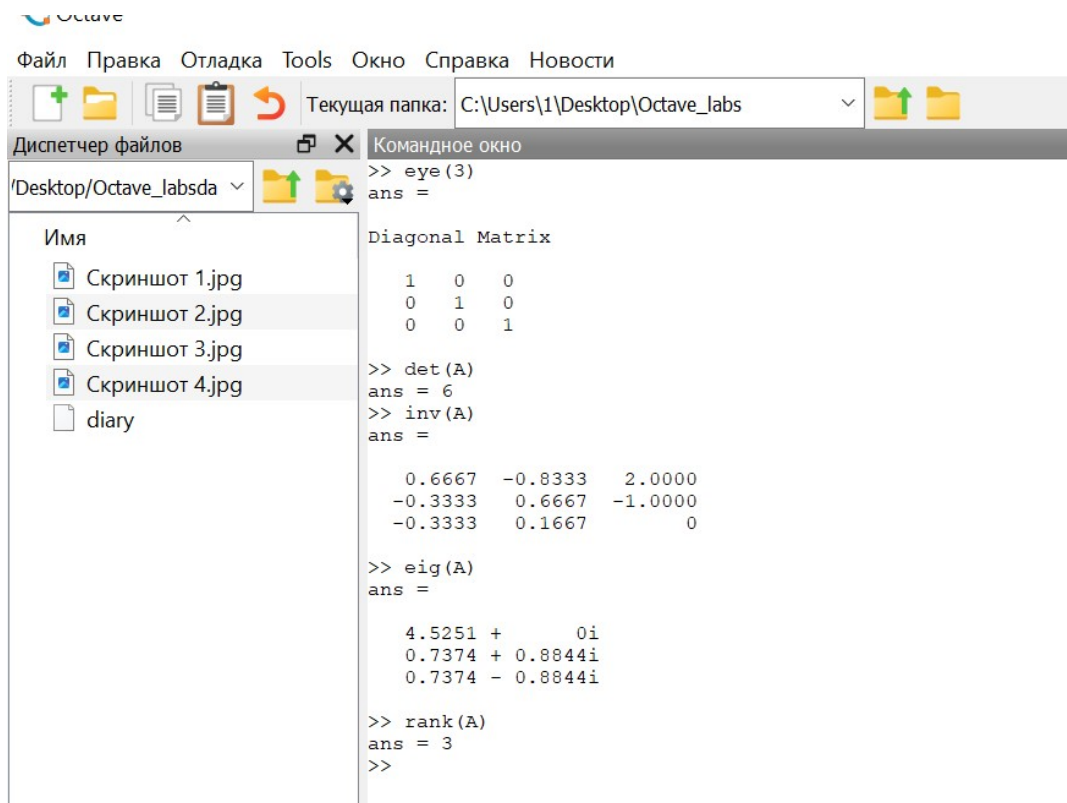


Рис. 3.5: Матричные операции

Создали вектор значений  $x$ . Задали вектор  $y = \sin(x)$  и построили график. Очистили рабочую область фигуры для дальнейшего улучшения графика. Задали красный цвет для линии и сделали ее потолще. Подогнали диапазон осей и нарисовали сетку. Подписали оси и сделали заголовок графика, а также задали легенду (рис. fig. 3.6 и рис. fig. 3.7).

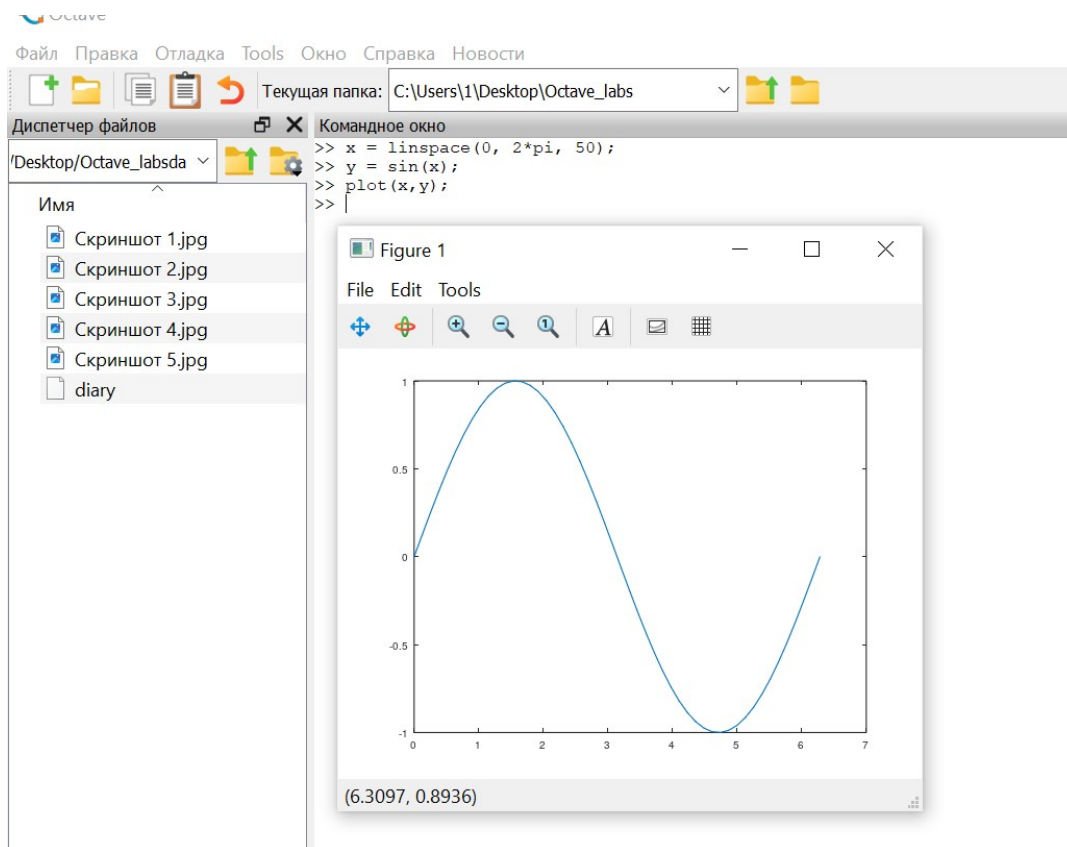


Рис. 3.6: Построение простейших графиков

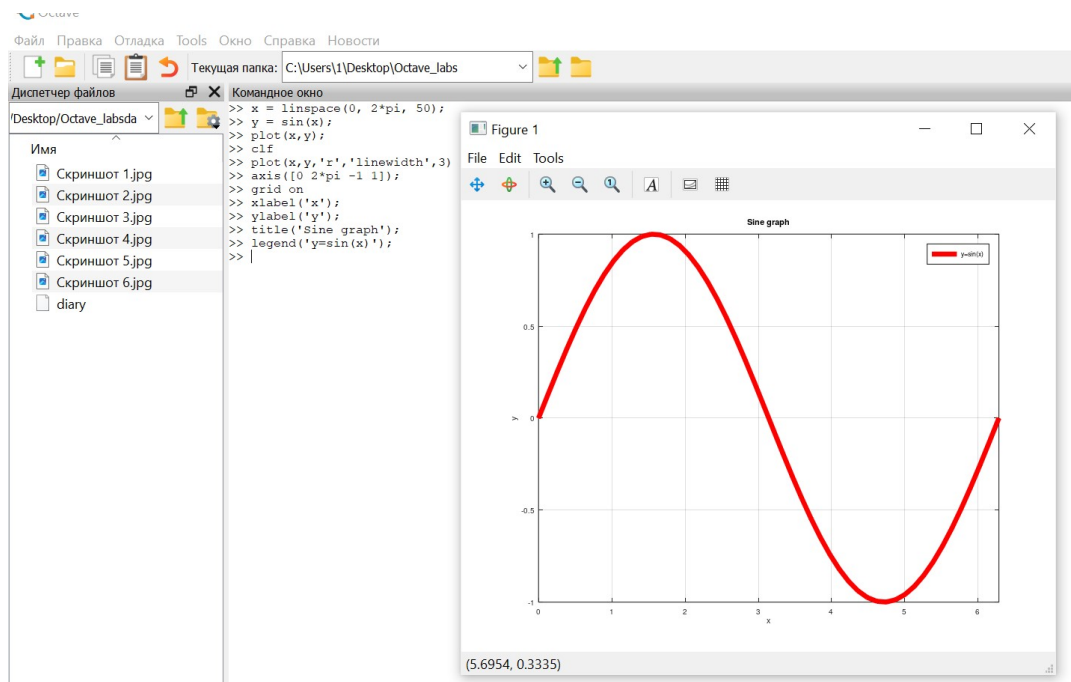


Рис. 3.7: Построение простейших графиков

Очистили память и рабочую область фигуры. Задали два вектора и начертили эти точки, используя кружочки как маркеры. Ввели команду для добавления еще одного графика к текущему. Добавили график регрессии. Задали сетку, оси и легенду (рис. fig. 3.8).

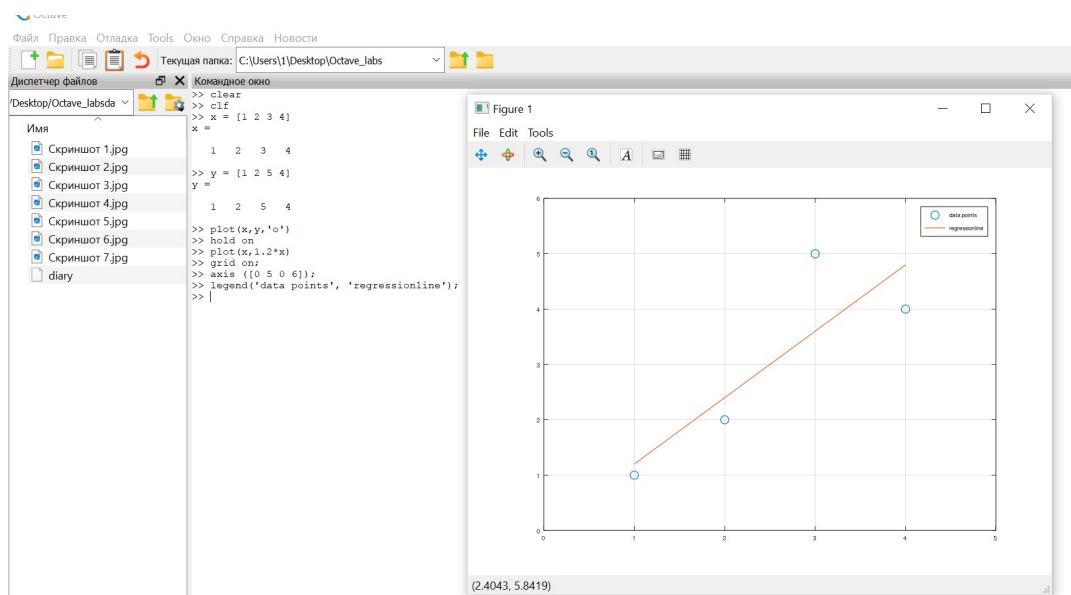


Рис. 3.8: Два графика на одном чертеже

Строим график  $y = x^2 \sin(x)$ . Очистим память и рабочую область фигуры. Зададим вектор  $x$  и построим график, используя поэлементное возведение в степень и поэлементное умножение (в противном случае выдаст ошибку). Сохраним графики в виде файлов (на сохранении файла в формате pdf программа зависает) (рис. fig. 3.9).

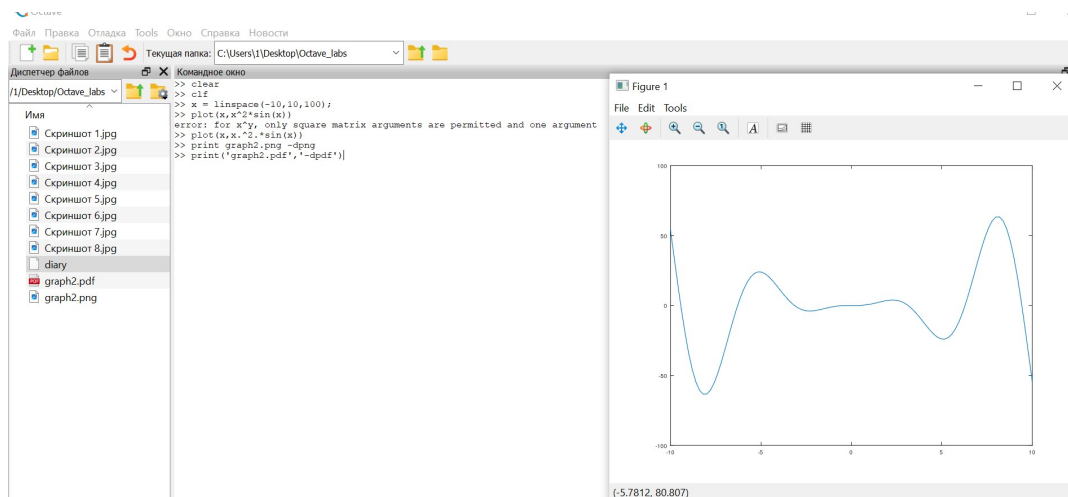


Рис. 3.9: График  $y = x^2 \sin(x)$

Сравним эффективность работы с циклами и операций с векторами. Для этого вычислим сумму:  $S = \sum_n^{100000} \frac{1}{n^2}$ .

Очистим память и рабочую область фигуры. Вычислим сумму  $S$  с помощью цикла, создав файл loop\_for.m (рис. fig. 3.10).

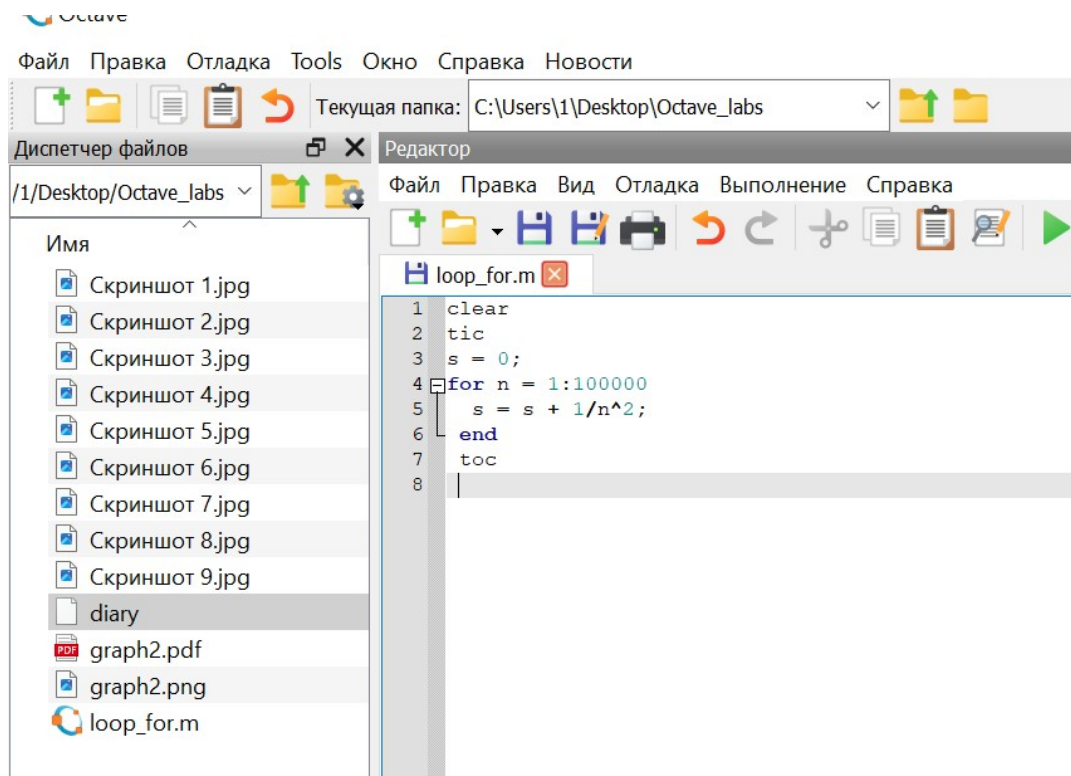


Рис. 3.10: Вычисление суммы с помощью цикла

Вычислим сумму  $S$  с помощью операций с векторами, создав файл `loop_vec.m` (рис. fig. 3.11).

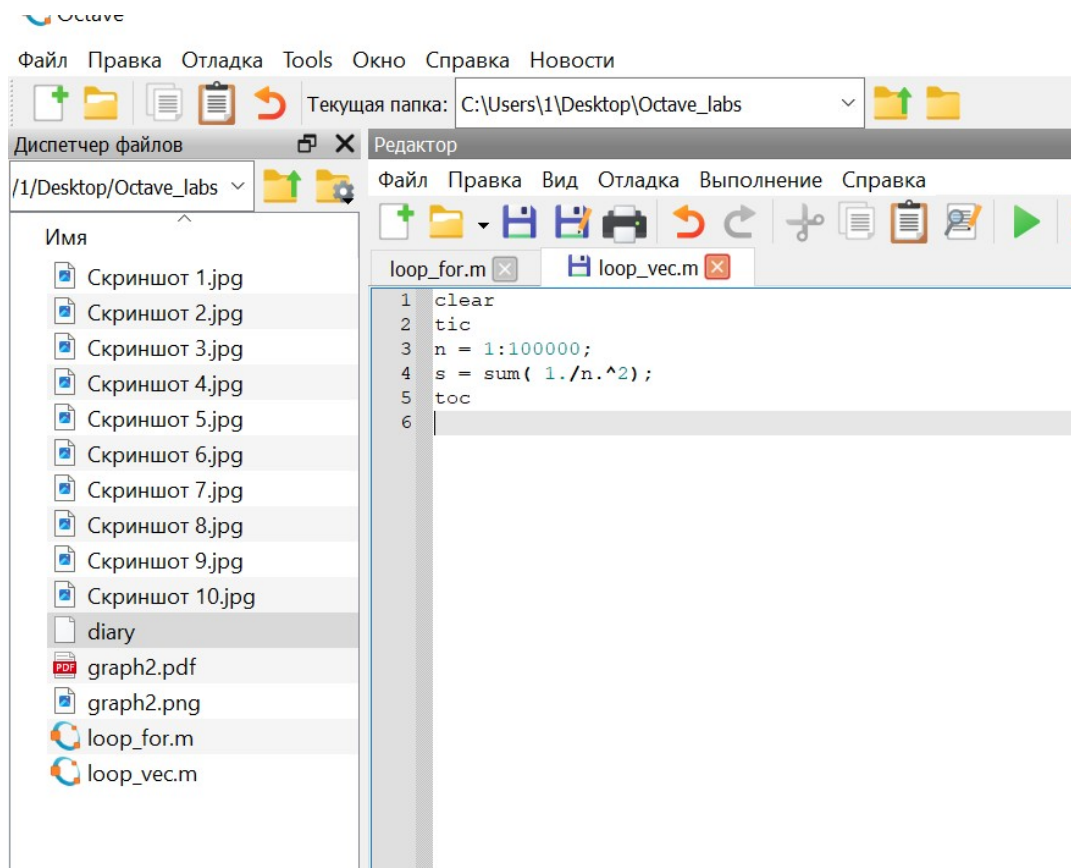


Рис. 3.11: Вычисление суммы с помощью операций с векторами

Запустив оба файла, видим, что время на выполнение второй программы в 100 раз меньше, чем на выполнение первой (рис. fig. 3.12). Таким образом, в данном случае операции с векторами эффективнее, чем циклы. В конце завершим запись в файл.

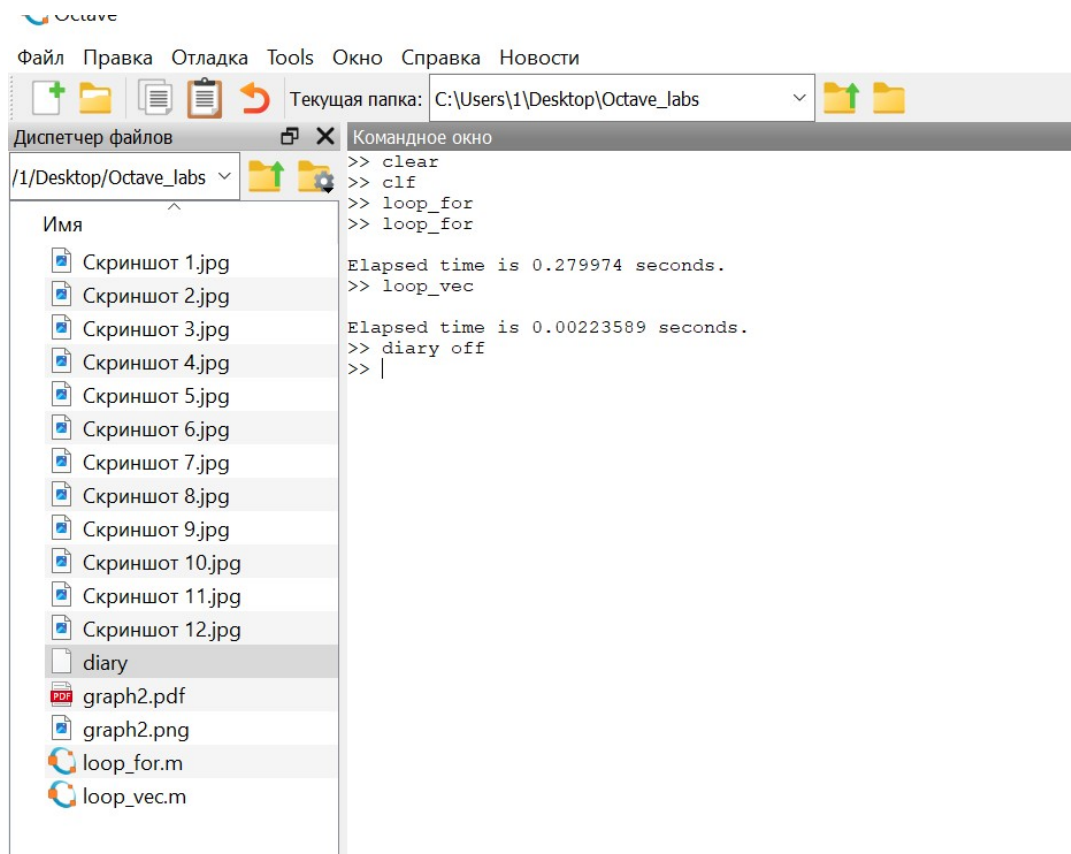


Рис. 3.12: Сравнение двух программ



## 4 Вывод

В ходе выполнения данной лабораторной работы я освоил базовые навыки работы в Octave: простейшие вычислительные операции, операции с векторами и матрицами, построение простейших графиков, сравнение циклов и операций с векторами.

## **Список литературы**