Лабораторная работа № 3

Управляющие структуры

Шияпова Дарина Илдаровна

Содержание

# 1 Цель работы

Основная цель работы — освоить применение циклов функций и сторонних для Julia пакетов для решения задач линейной алгебры и работы с матрицами.

# 2 Задание

1. Используя Jupyter Lab, повторите примеры из раздела 3.2.
2. Выполните задания для самостоятельной работы (раздел 3.4)

# 3 Теоретическое введение

Julia – высокоуровневый свободный язык программирования с динамической типизацией, созданный для математических вычислений [**julialang?**]. Эффективен также и для написания программ общего назначения. Синтаксис языка схож с синтаксисом других математических языков, однако имеет некоторые существенные отличия.

Для выполнения заданий была использована официальная документация Julia [**juliadoc?**].

# 4 Выполнение лабораторной работы

Для начала выполним примеры из лабораторной работы, чтобы познакомиться с циклами, условными операторами, функциями и работой со сторонними библиотеками (рис. 1-3).

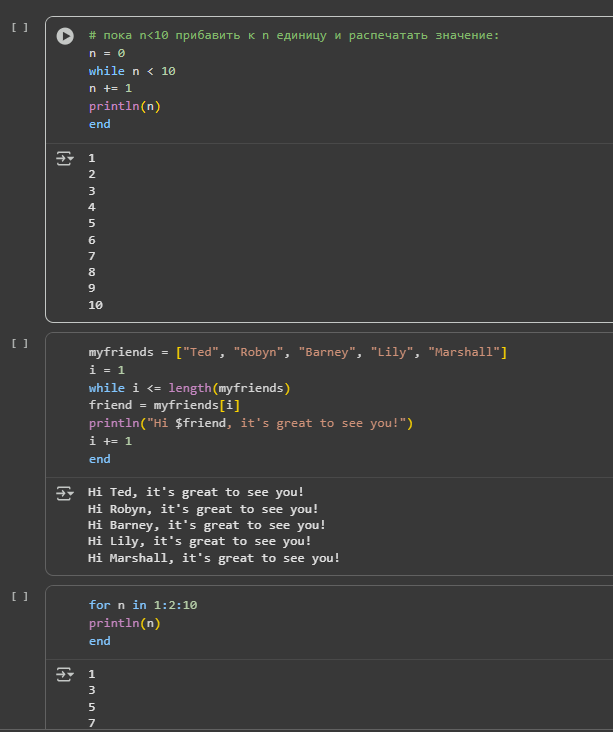


Рис. 1: Выполнение примеров с циклами

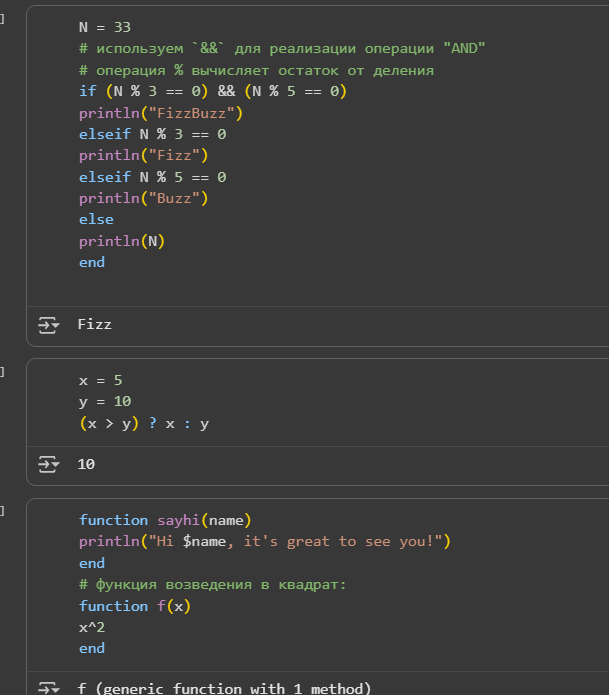


Рис. 2: Выполнение примеров с условными выражениями

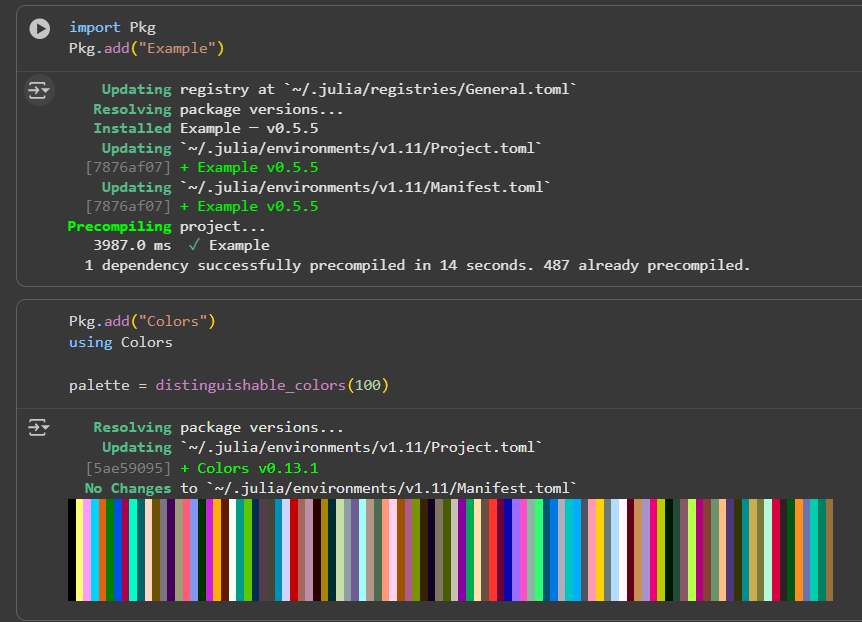


Рис. 3: Выполнение примеров со сторонними библиотеками

Теперь перейдем к выполнению заданий для самостоятельной работы.

Используя циклы while и for (рис. 4-6):

* выведем на экран целые числа от 1 до 100 и напечатаем их квадраты;
* создадим словарь squares, который будет содержать целые числа в качестве ключей и квадраты в качестве их пар-значений;
* создадим массив squares\_arr, содержащий квадраты всех чисел от 1 до 100.

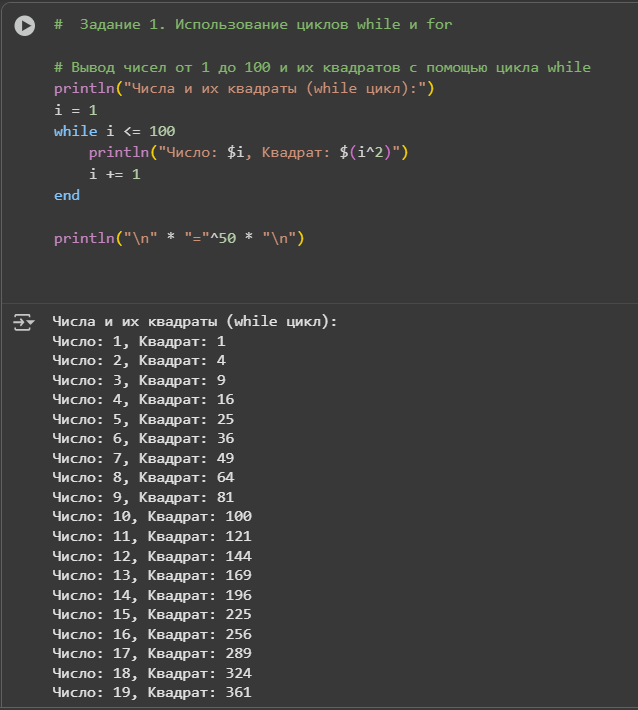


Рис. 4: Задание №1

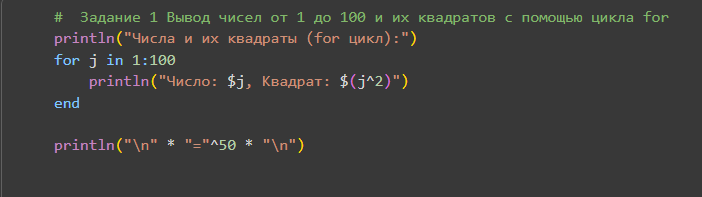


Рис. 5: Задание №1

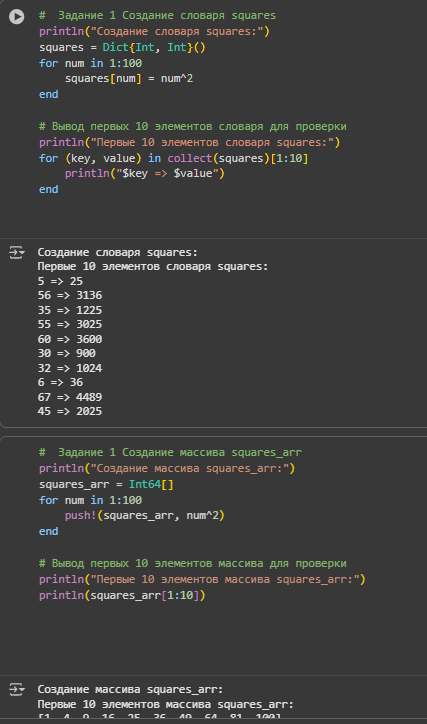


Рис. 6: Задание №1

Напишем условный оператор, который печатает число, если число чётное, и строку «нечётное», если число нечётное. Перепишем код, используя тернарный оператор (рис. 7).

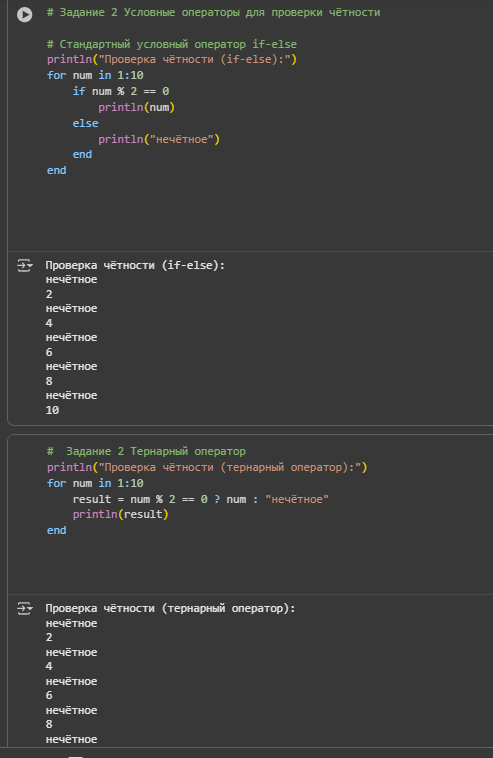


Рис. 7: Задание №2

Напишем функцию add\_one, которая добавляет 1 к своему входу (рис. 8).

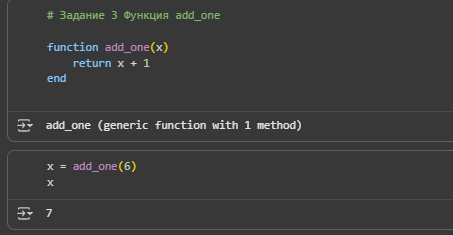


Рис. 8: Задание №3

Используем map() или broadcast() для задания матрицы 𝐴, каждый элемент которой увеличивается на единицу по сравнению с предыдущим. (рис. 9)

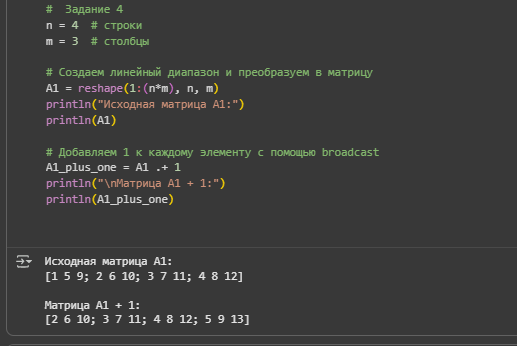


Рис. 9: Задание №4

Зададим матрицу A. Найдем A^3. Заменим третий столбец матрицы 𝐴 на сумму второго и третьего столбцов (рис. 10).

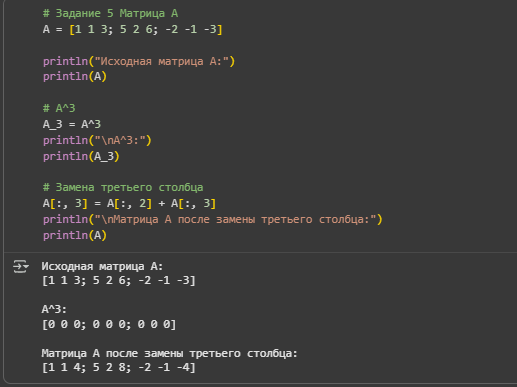


Рис. 10: Задание №5

Задание 6 (рис. 11)

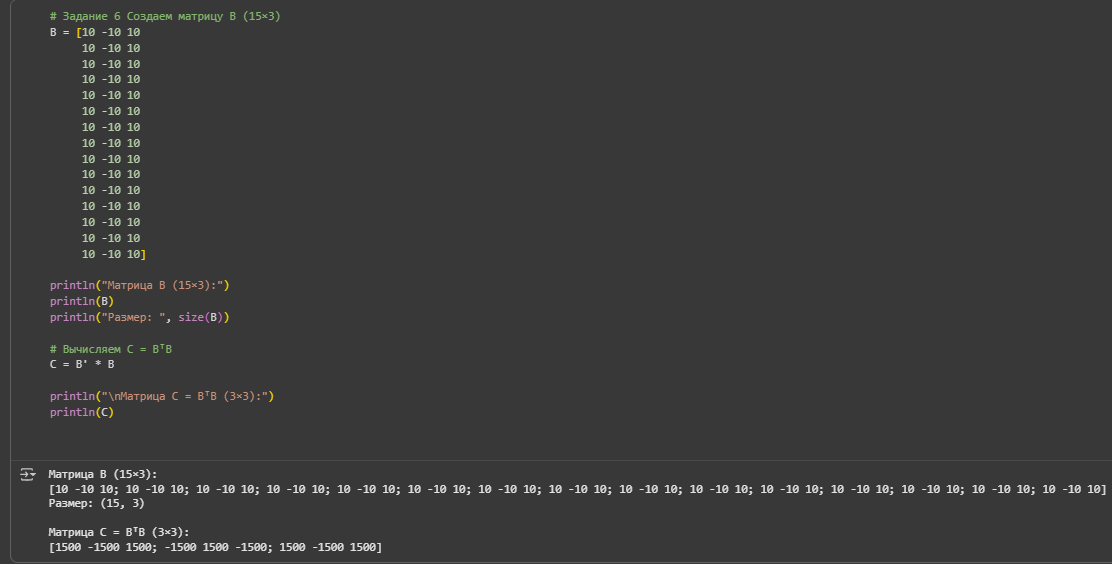


Рис. 11: Задание №6

Задание 7 (рис. 12)



Рис. 12: Задание №7

Напишем свою функцию, аналогичную функции outer() языка R. Функция должна иметь следующий интерфейс: outer(x,y,operation) (рис. 13,14).

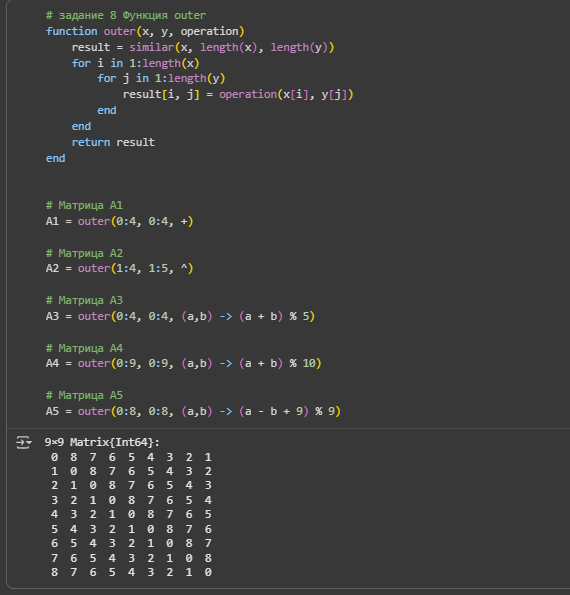


Рис. 13: Реализация функции outer()

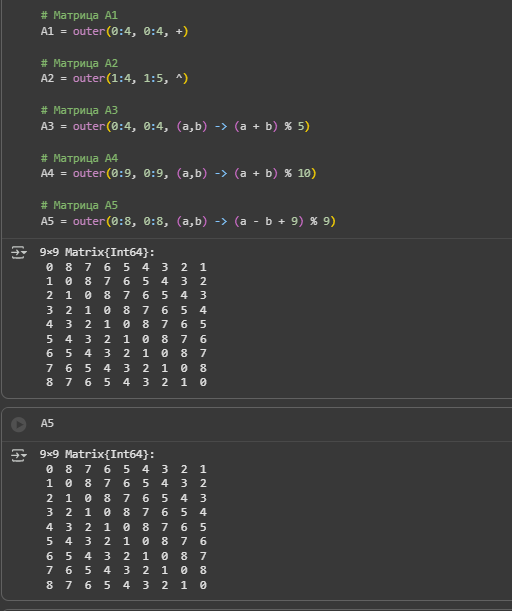


Рис. 14: Проверка работы функции outer()

Решим систему линейных уравнений с 5 неизвестными (рис. 15).



Рис. 15: Решение систему линейных уравнений

В 10 задании произведем анализ количества элементов матрицы, удовлетворяющих необходимым условиям (рис. 16).



Рис. 16: Задание №10

Вычислим выражения (рис. 17).



Рис. 17: Задание №11

# 5 Выводы

В результате выполнения данной лабораторной работы я освоила применение циклов функций и сторонних для Julia пакетов для решения задач линейной алгебры и работы с матрицами.

# Список литературы