Отчёт по лабораторной работе №3

Компьютерный практикум по статистическому анализу данных

Канева Екатерина, НФИбд-02-22

Содержание

Список иллюстраций

Список таблиц

# 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

* Используя Jupyter Lab, повторить примеры.
* Выполнить задания для самостоятельной работы.

# 3 Теоретическая часть

Julia - высокоуровневый свободный язык программирования с динамической типизацией, созданный для математических вычислений. 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.

# 4 Выполнение лабораторной работы

Сначала я выполнила примеры с циклами while и for (рис. 1, 2):



Рис. 1: Примеры с циклами (1).

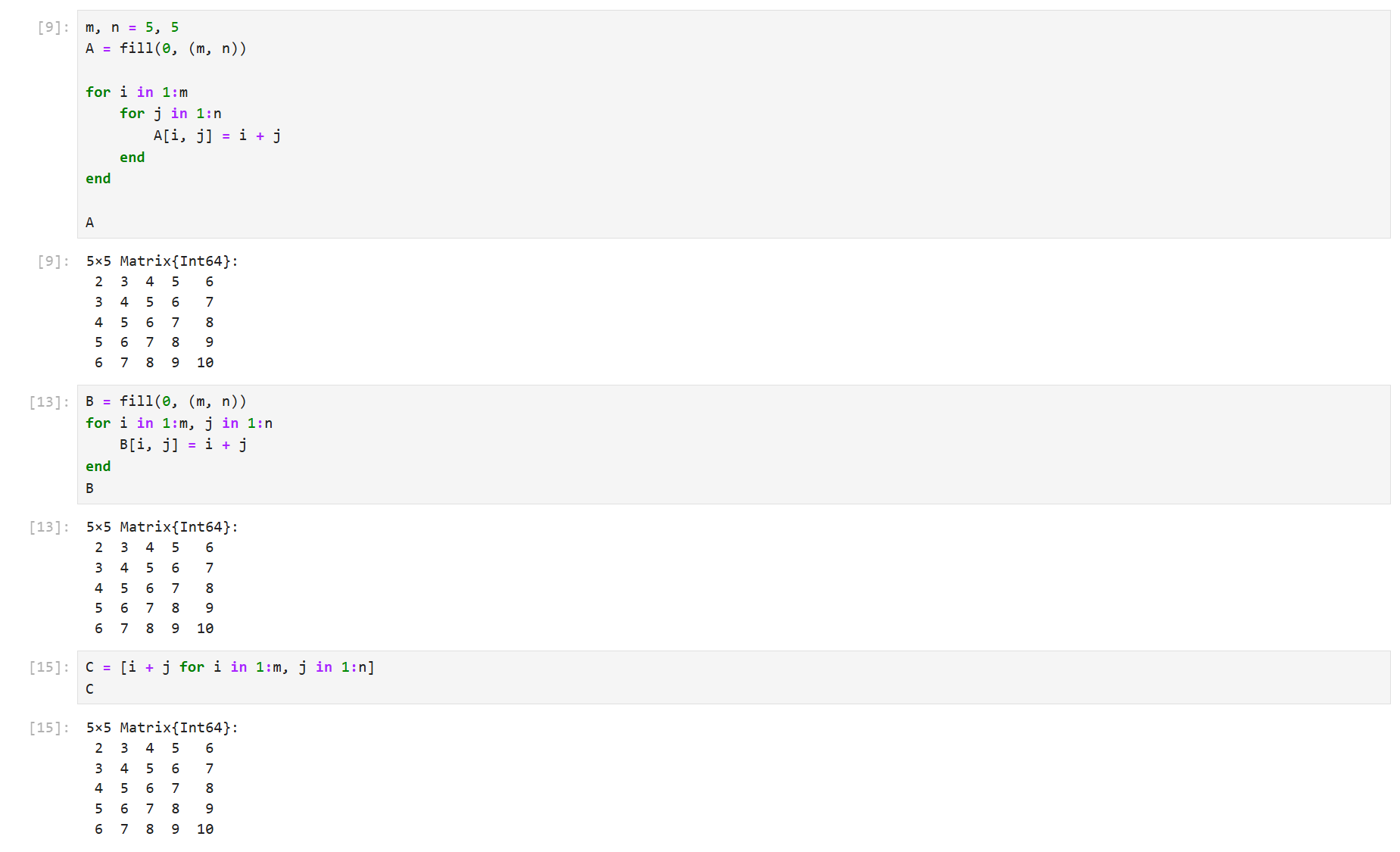


Рис. 2: Примеры с циклами (2).

Потом я выполнила примеры с условными выражениями (рис. 3):

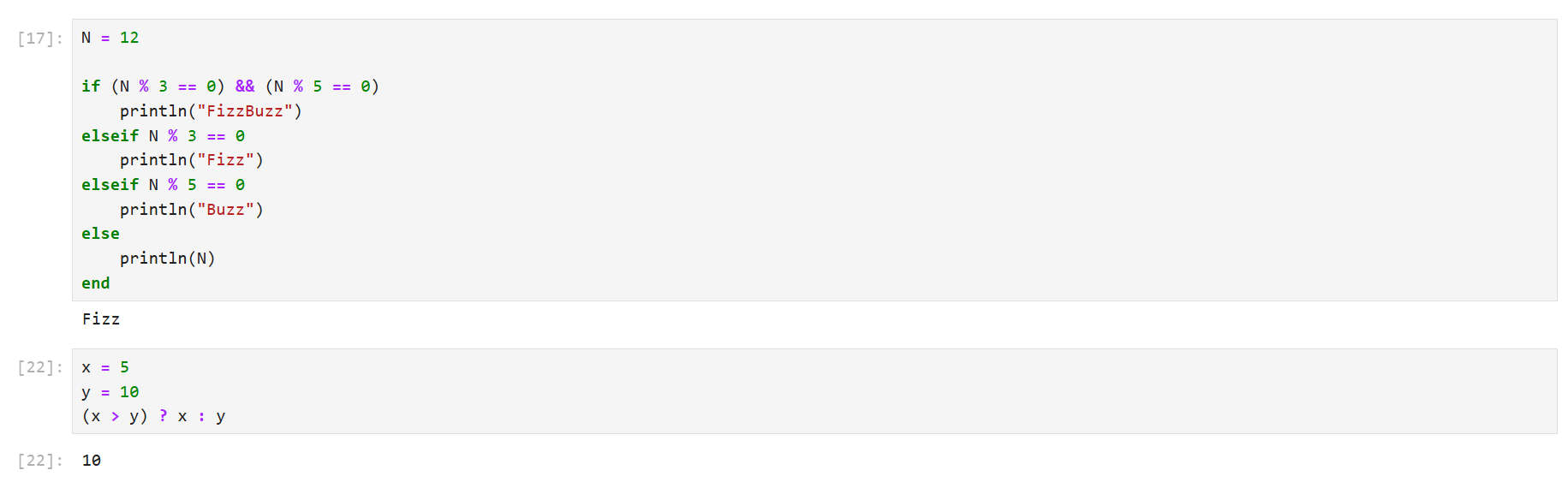


Рис. 3: Примеры с условными выражениями.

Потом я выполнила примеры с функциями (рис. 4):

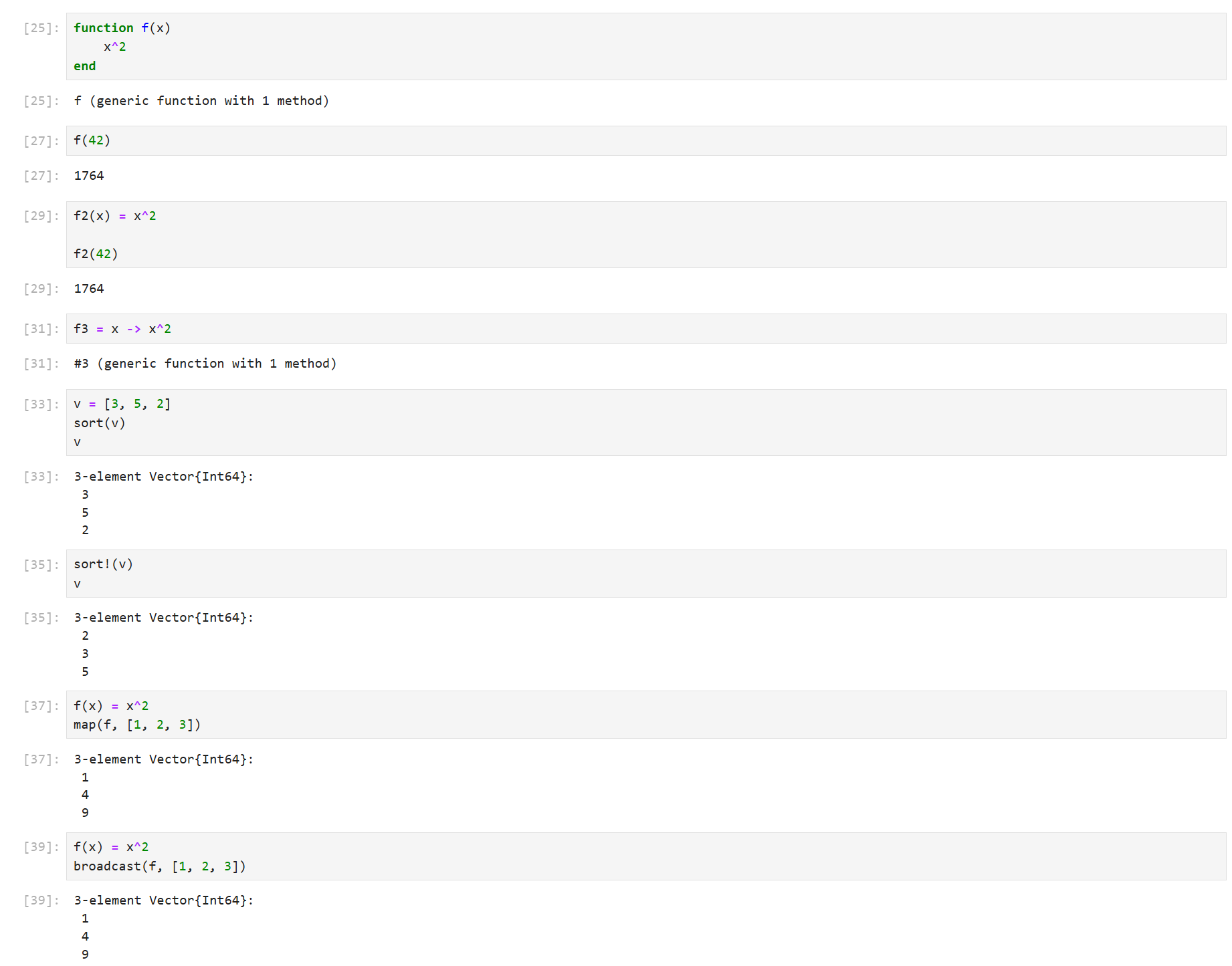


Рис. 4: Примеры с функциями.

Потом я выполнила примеры со сторонними библиотеками (рис. 5):

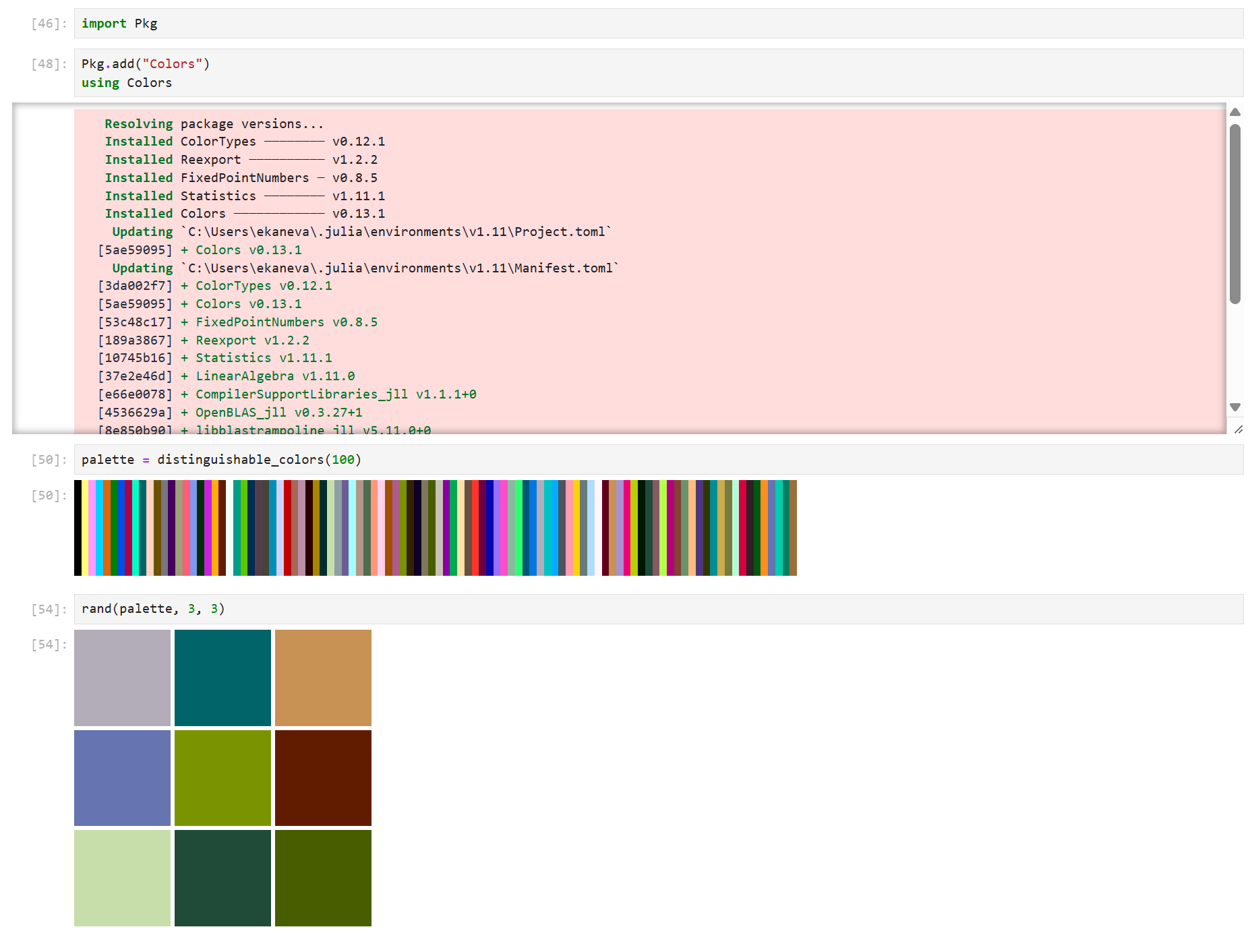


Рис. 5: Примеры со сторонними библиотеками.

Далее я приступила к выполнению заданий для самостоятельной работы.

1. Используя циклы while и for.

* 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.

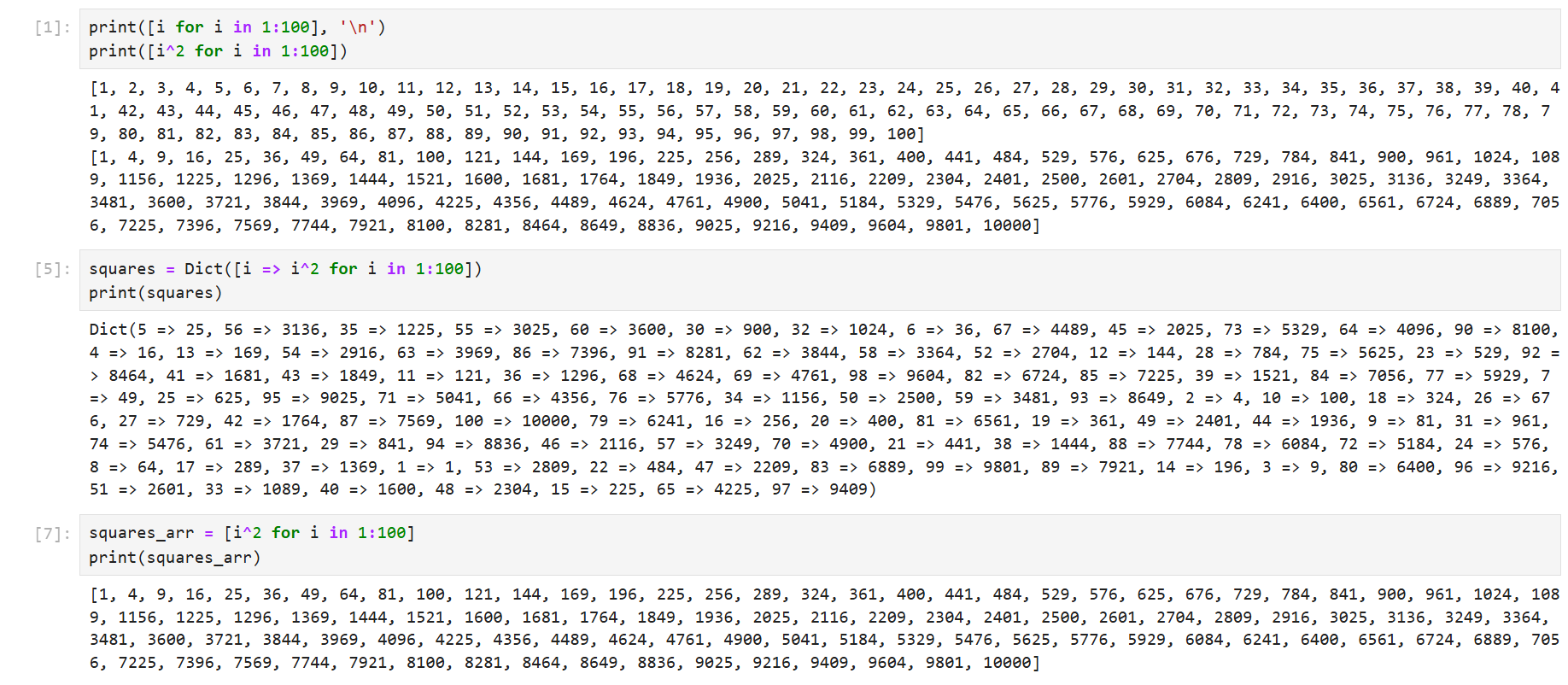


Рис. 6: Задание 1.

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

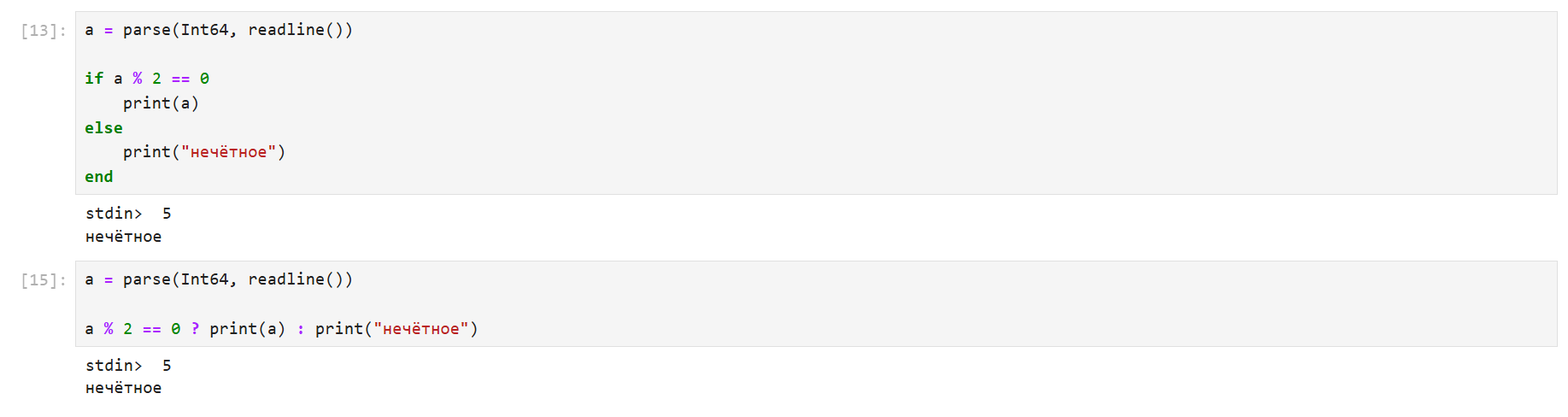


Рис. 7: Задание 2.

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

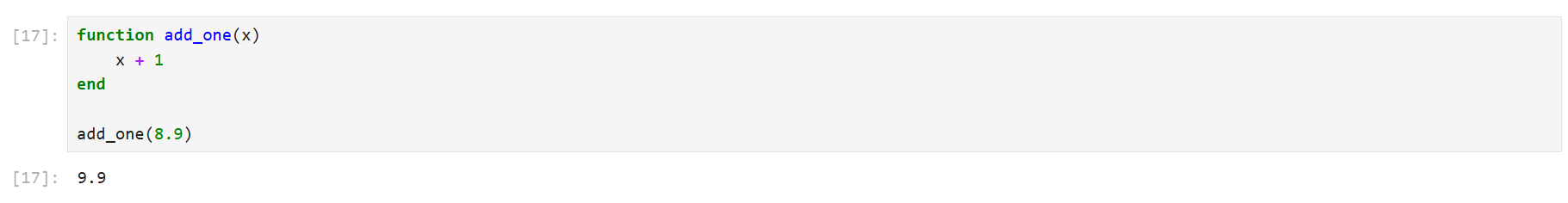


Рис. 8: Задание 3.

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

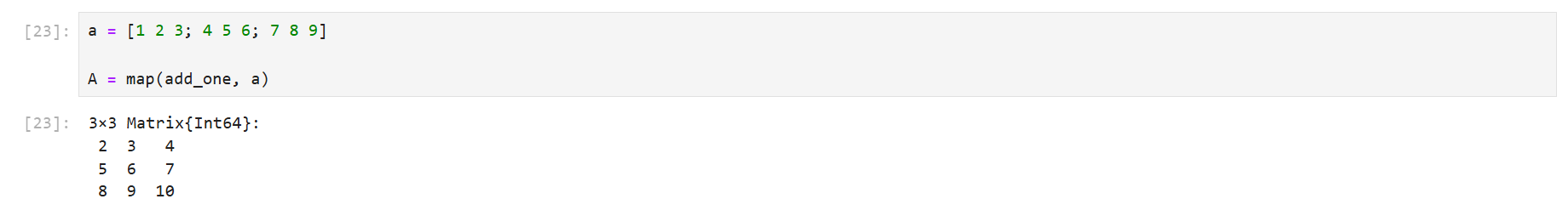


Рис. 9: Задание 4.

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

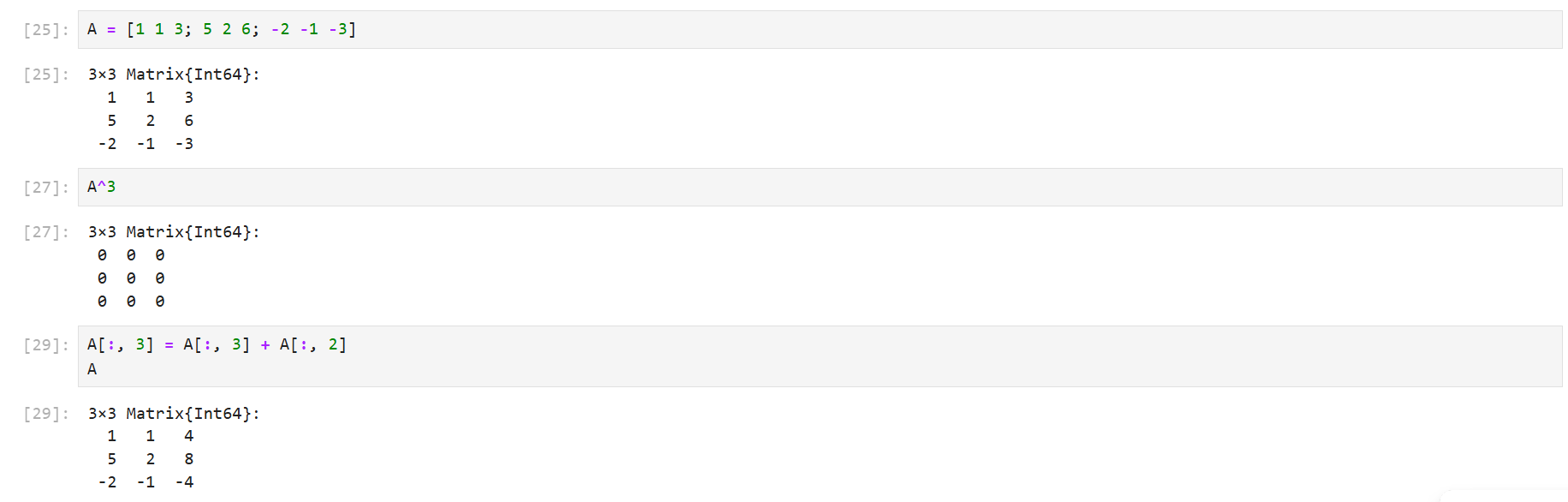


Рис. 10: Задание 5.

1. Создадим матрицу B с элементами , , , . Вычислим матрицу . (рис. 11).

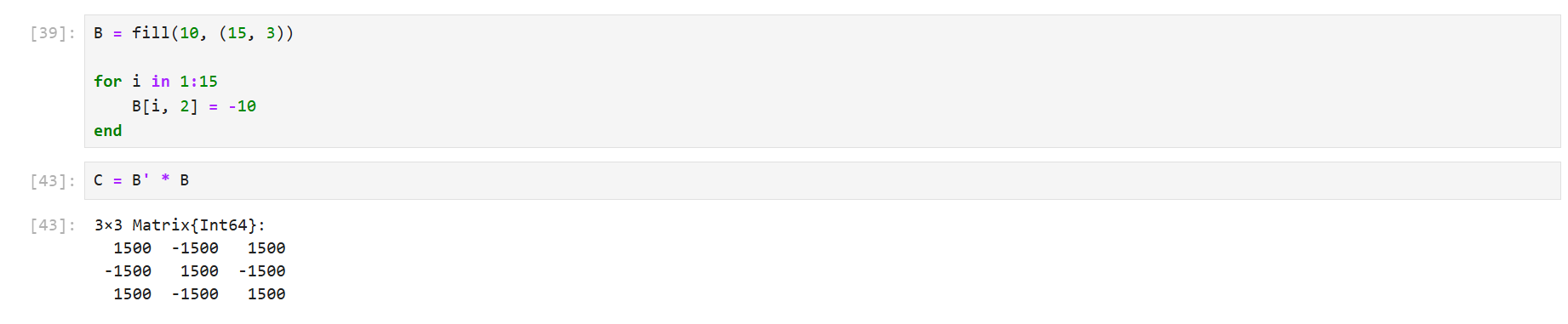


Рис. 11: Задание 6.

1. Создадим матрицу Z размерности 6x6, все элементы которой равны нулю, и матрицу E, все элементы которой равны 1. Используя цикл while или for и закономерности расположения элементов, создадим новые матрицы размерности 6x6 (рис. 12-13):

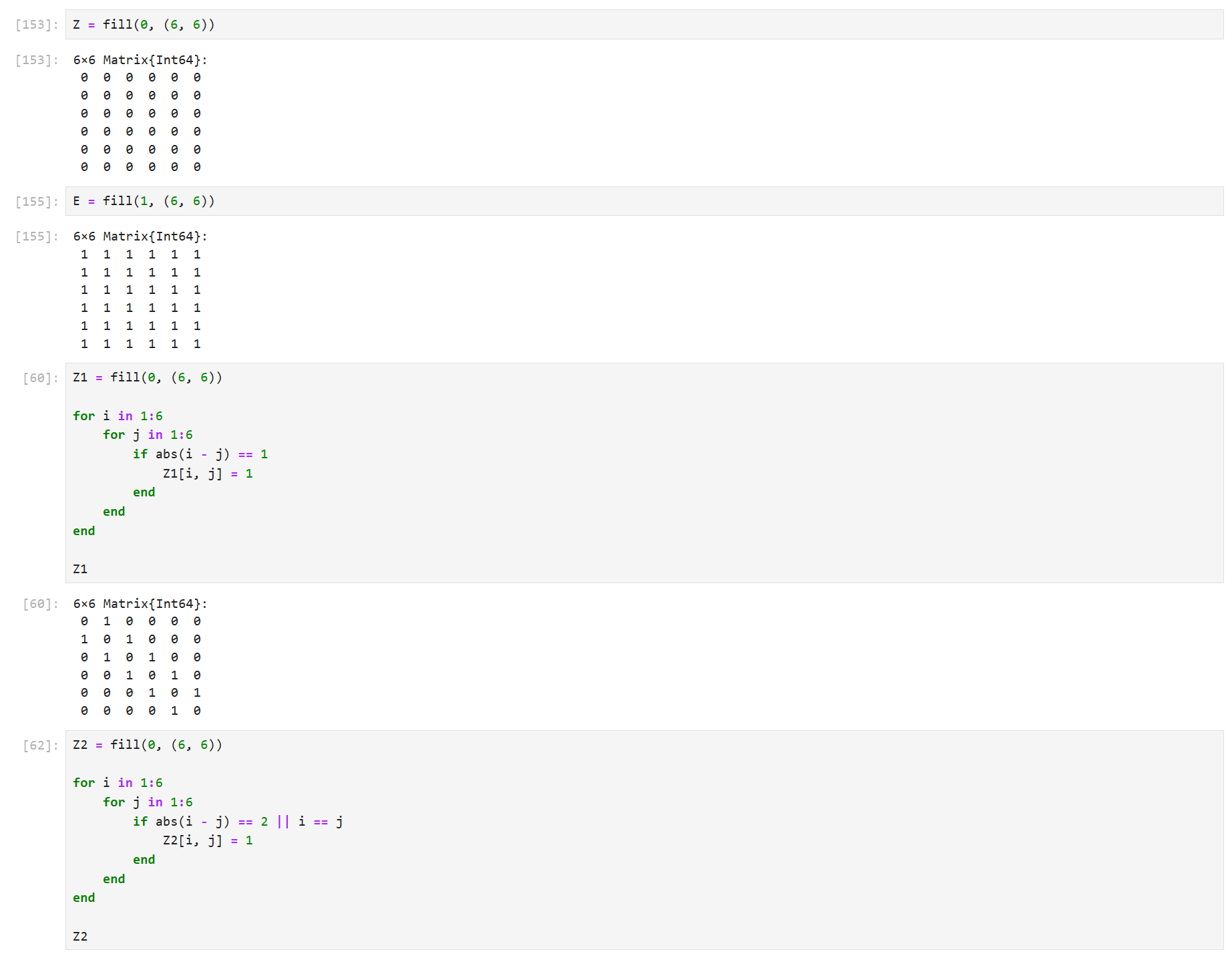


Рис. 12: Задание 7(1).

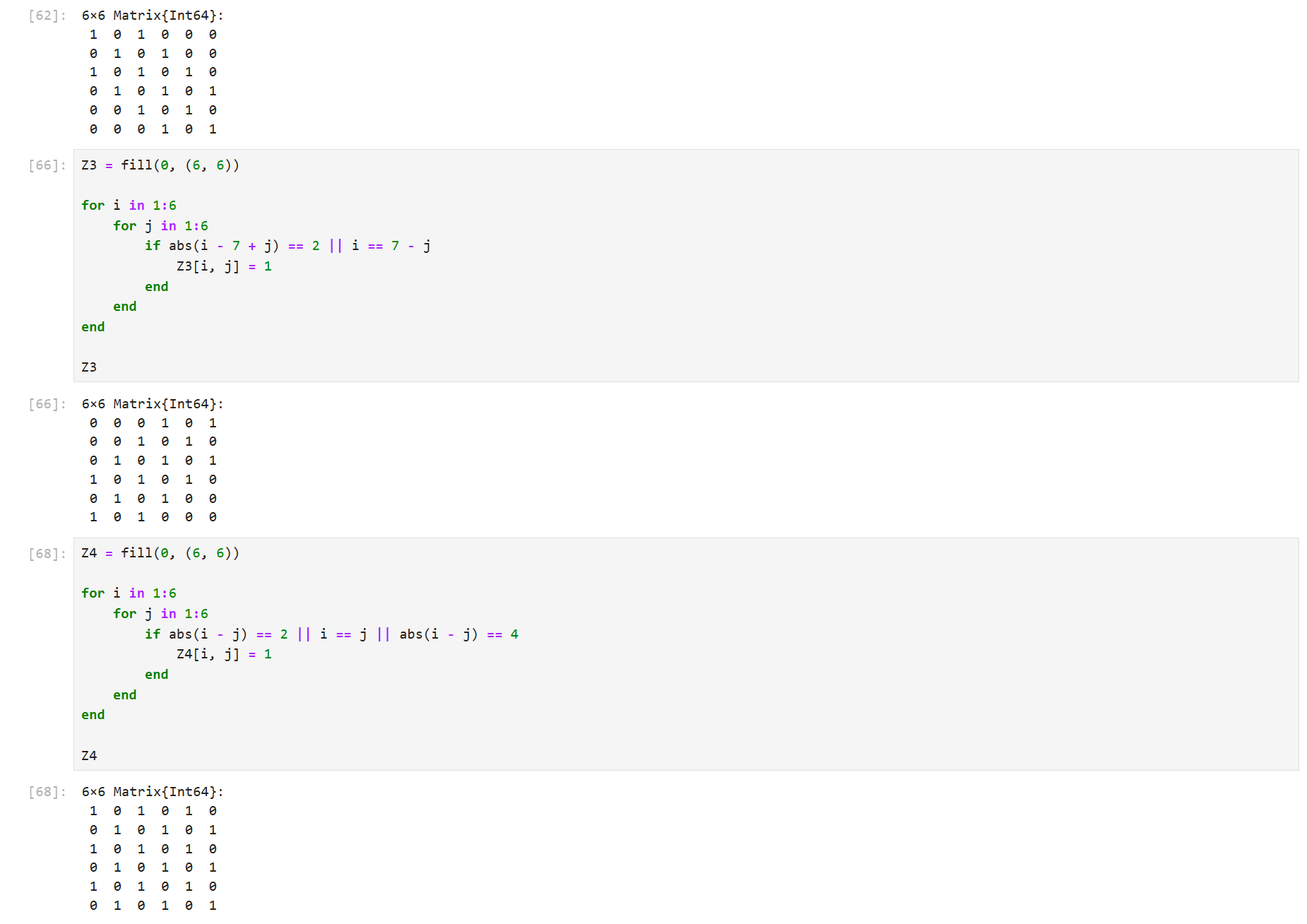


Рис. 13: Задание 7(2).

1. В языке R есть функция outer(). Напишем свою функцию, аналогичную функции outer() языка R. Функция будет иметь следующий интерфейс: outer(x,y,operation). Используя написанную функцию outer(), создадим новые матрицы (рис. 14-15):

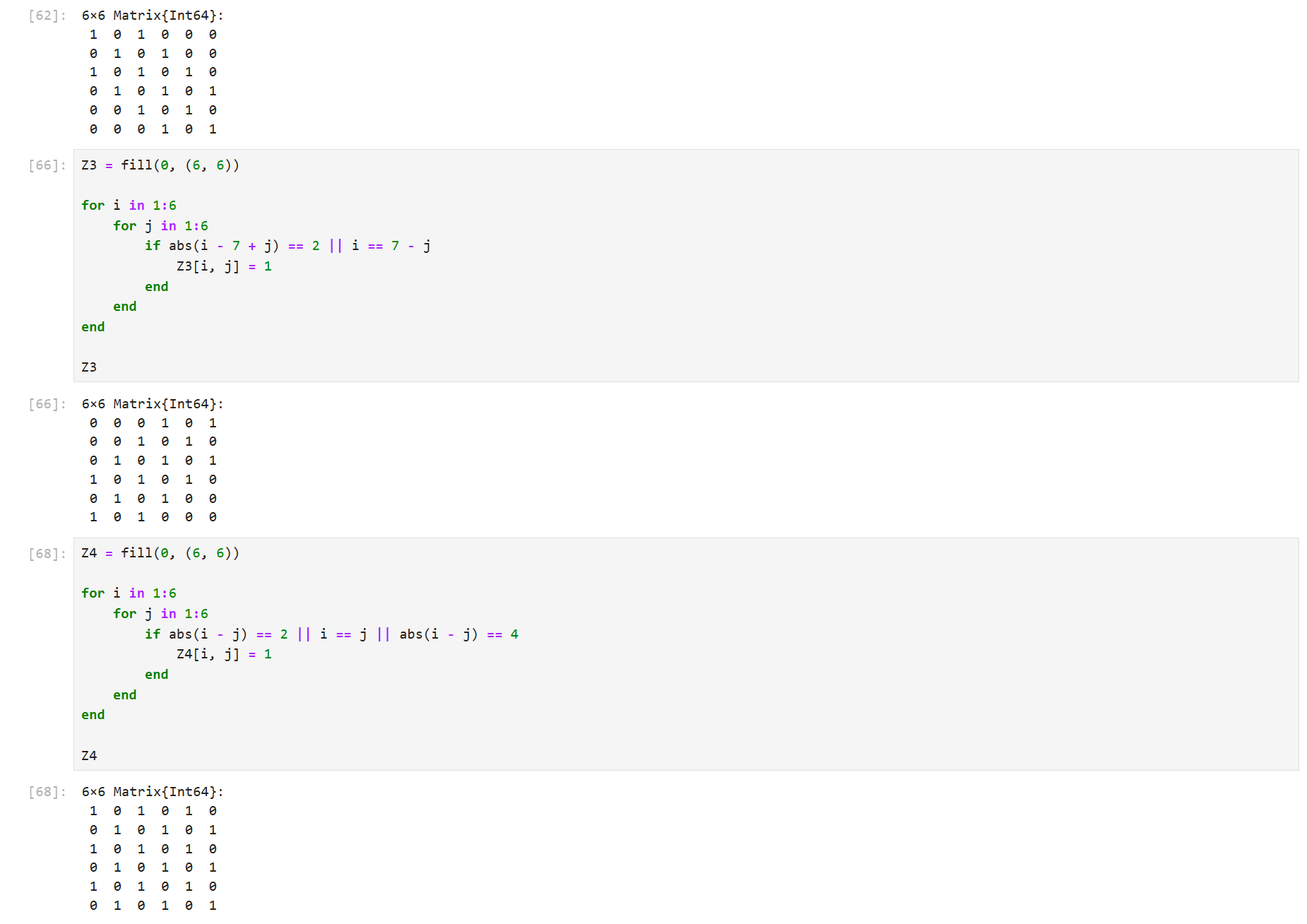


Рис. 14: Задание 8(1).

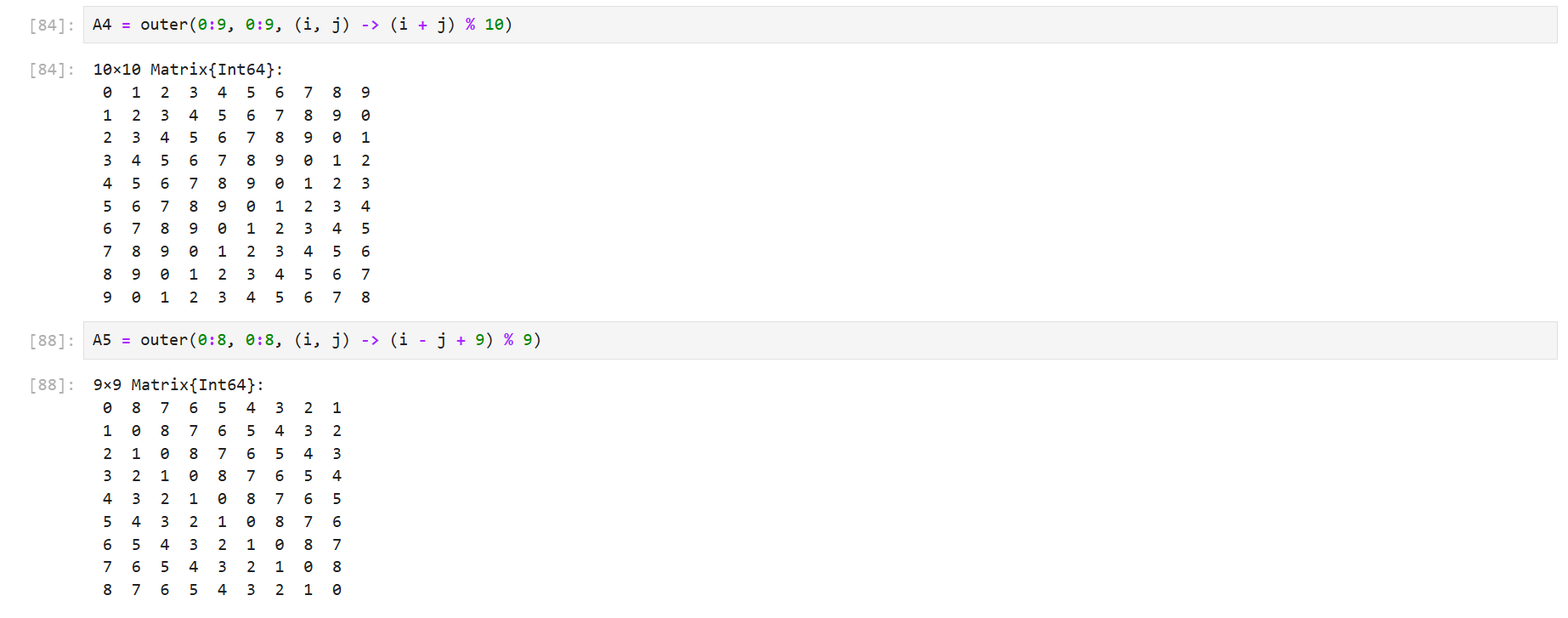


Рис. 15: Задание 8(2).

1. Решим систему линейных уравнений с 5 неизвестными, рассмотрев соответствующее матричное уравнение (рис. 10):

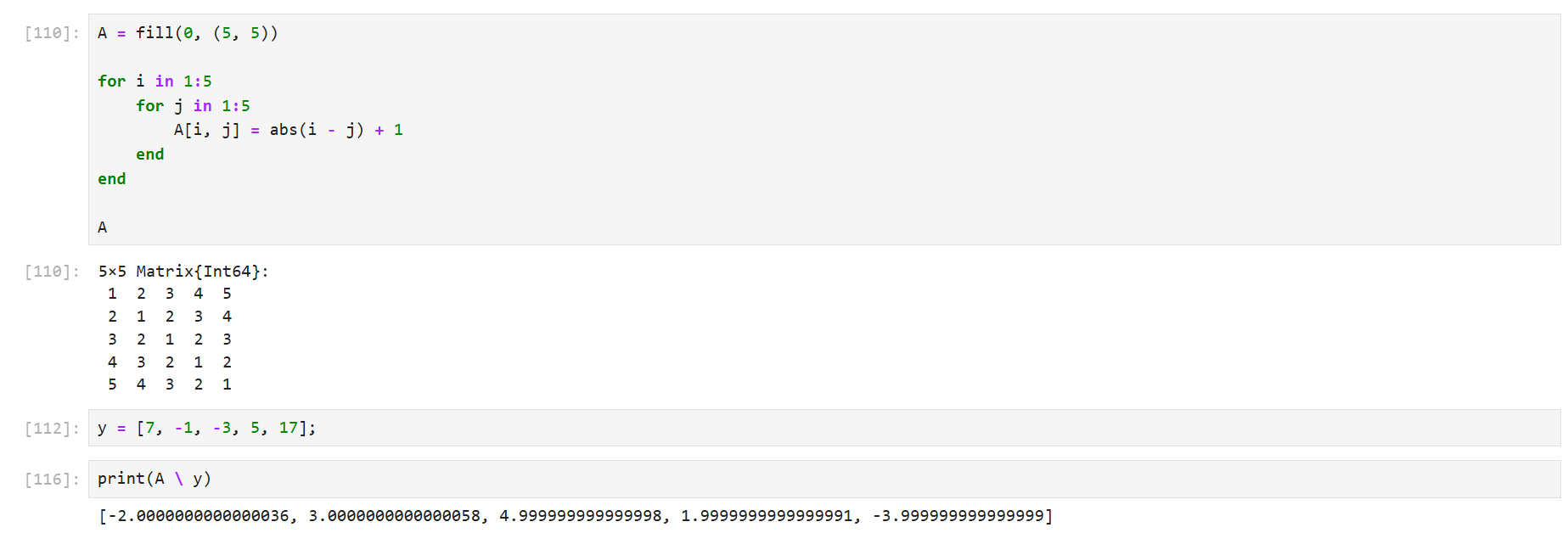


Рис. 16: Задание 9.

1. Создадим матрицу M размерности 6x10, элементами которой являются целые числа, выбранные случайным образом с повторениями из совокупности 1-10. Найдём число элементов в каждой строке матрицы M, которые больше числа N (например, N = 4), определим, в каких строках матрицы M число M (например, M = 7) встречается ровно 2 раза, определим все пары столбцов матрицы M, сумма элементов которых больше K (например, K = 75) (рис. 10):

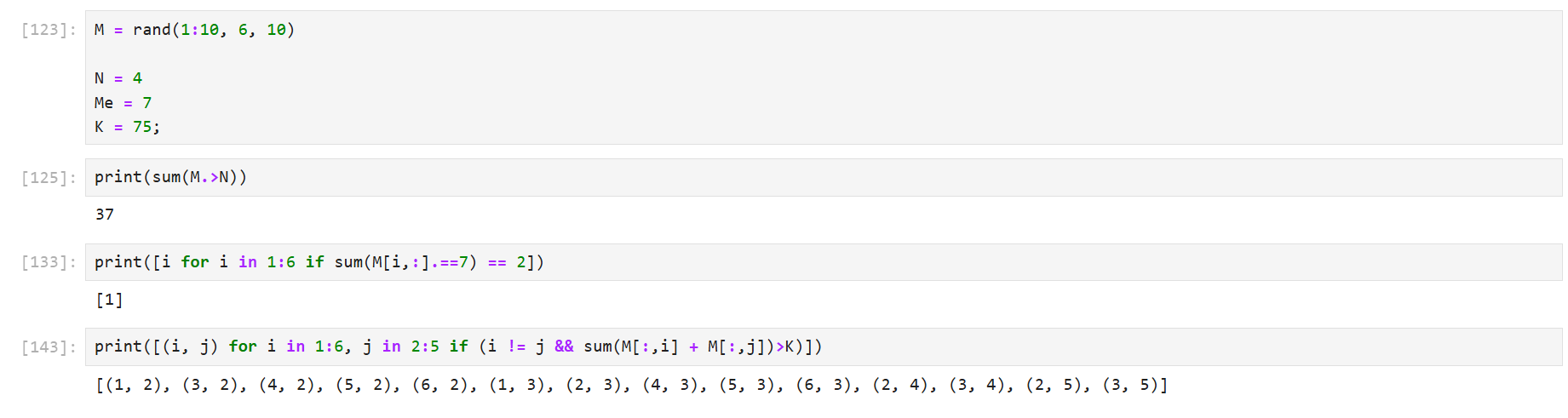


Рис. 17: Задание 10.

1. Вычислим суммы (рис. 10):

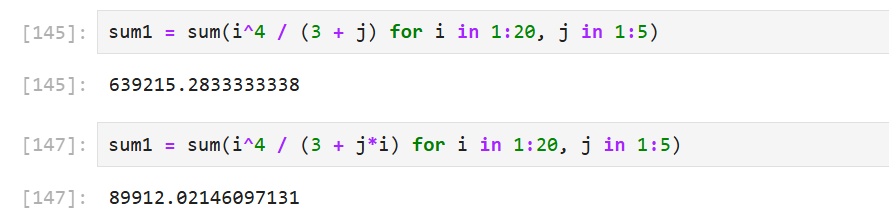


Рис. 18: Задание 11.

# 5 Выводы

Освоила применение циклов функций и сторонних для Julia пакетов для решения задач линейной алгебры и работы с матрицами.