Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

**Отчет по лабораторной работе №2**

Дисциплина: «Разработка профессиональных приложений»

Двумерные структуры данных

Вариант №9

Выполнил:

студент группы ИВТАПбд-22

Ступин А. О.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

**Задание по варианту**

Выполнить обработку элементов прямоугольной матрицы A, имеющей N строк и M столбцов. Определить, сколько нулевых элементов содержится в верхних L строках матрицы и в левых К столбцах матрицы.

**Требования к программе:**

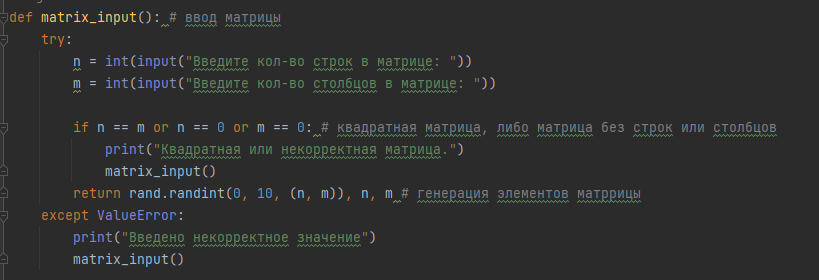
1. Ввод элементов списка должен быть доступен путем автоматической генерации. Необходимо использовать библиотеку numpy. Результаты выполнения должны сохраняться в файл (исходные данные и результат обработки).

2. Исходный код должен быть откомментирован

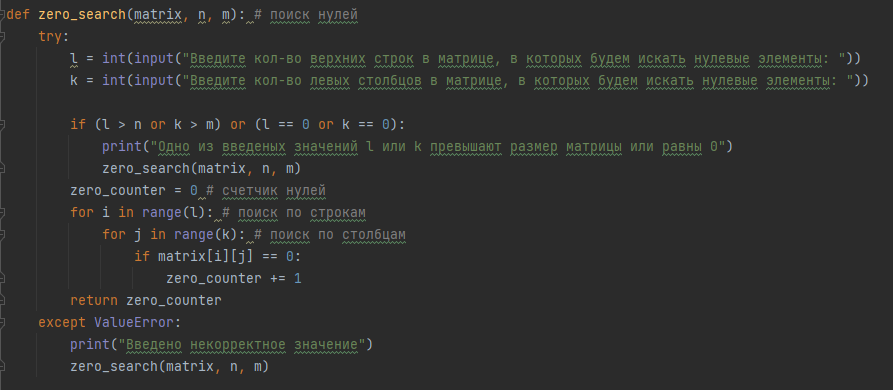
3. Необходимо реализовать правильную декомпозицию программы на методы.

**Особенности реализации**

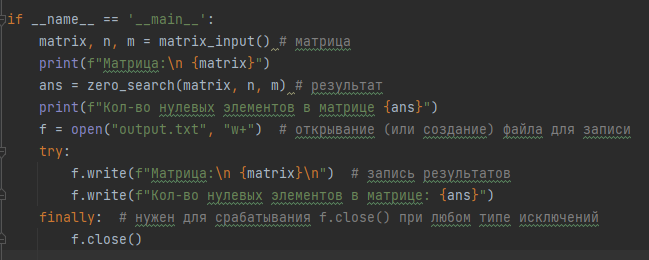
Метод matrix\_input отвечает за ввод размеров и генерирование матрицы. Матрица создаётся из рандомных элементов от нуля до 10 как двумерной массив размерностью в l строк и k столбцов.



Метод zero\_search отвечает за поиск нулей. В нём производится проверка каждого элемента в введенных пользователем границах на равенство нулю: если такой элемент находится, то к счётчику прибавляется единица. В конечном итоге этот счетчик и будет возвращать ответ, какого же количество нулей.



В функции main вызываются два предыдущих метода, матрица и количество нулей выводятся в консоль и записываются в текстовый файл.



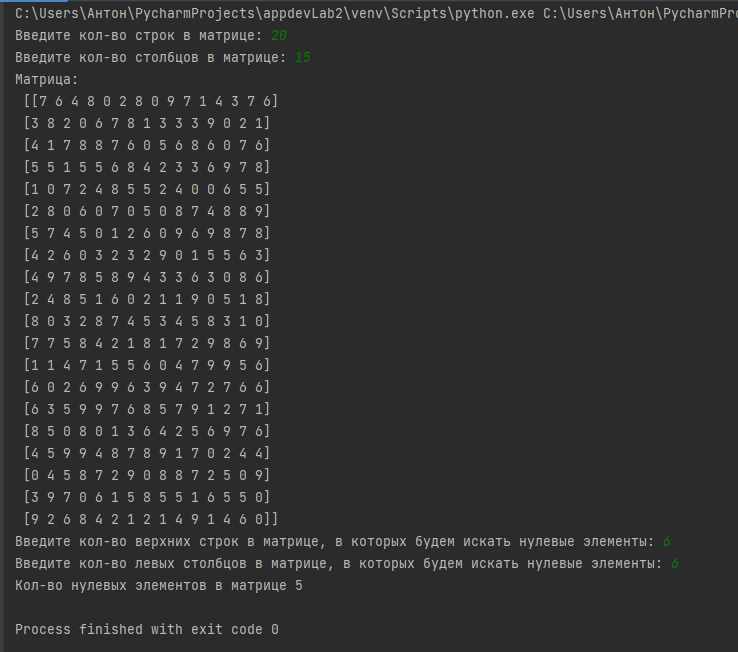
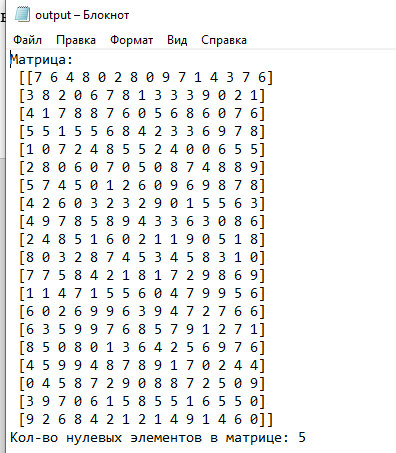
**Описание возникших затруднений**

При выполнении лабораторной работы серьёзных затруднений не возникло.

**Альтернативные способы решения**

Можно было реализовать задание более мудрено: сгенерированную матрицу сохранить в файл input, затем в методе zero\_search открыть этот файл на чтение и высчитывать нули уже из него. Затем в output сохранить ответ, предварительно скопировав в него содержимое файла input, то есть матрицу.

**Тестирование программы**



**Приложение 1. Исходный код программы**

import numpy as np  
import numpy.random as rand  
  
  
def matrix\_input(): # ввод матрицы  
 try:  
 n = int(input("Введите кол-во строк в матрице: "))  
 m = int(input("Введите кол-во столбцов в матрице: "))  
  
 if n == m or n == 0 or m == 0: # квадратная матрица, либо матрица без строк или столбцов  
 print("Квадратная или некорректная матрица.")  
 matrix\_input()  
 return rand.randint(0, 10, (n, m)), n, m # генерация элементов матррицы  
 except ValueError:  
 print("Введено некорректное значение")  
 matrix\_input()  
  
  
def zero\_search(matrix, n, m): # поиск нулей  
 try:  
 l = int(input("Введите кол-во верхних строк в матрице, в которых будем искать нулевые элементы: "))  
 k = int(input("Введите кол-во левых столбцов в матрице, в которых будем искать нулевые элементы: "))  
  
 if (l > n or k > m) or (l == 0 or k == 0):  
 print("Одно из введеных значений l или k превышают размер матрицы или равны 0")  
 zero\_search(matrix, n, m)  
 zero\_counter = 0 # счетчик нулей  
 for i in range(l): # поиск по строкам  
 for j in range(k): # поиск по столбцам  
 if matrix[i][j] == 0:  
 zero\_counter += 1  
 return zero\_counter  
 except ValueError:  
 print("Введено некорректное значение")  
 zero\_search(matrix, n, m)  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 matrix, n, m = matrix\_input() # матрица  
 print(f"Матрица:\n {matrix}")  
 ans = zero\_search(matrix, n, m) # результат  
 print(f"Кол-во нулевых элементов в матрице {ans}")  
 f = open("output.txt", "w+") # открывание (или создание) файла для записи  
 try:  
 f.write(f"Матрица:\n {matrix}\n") # запись результатов  
 f.write(f"Кол-во нулевых элементов в матрице: {ans}")  
 finally: # нужен для срабатывания f.close() при любом типе исключений  
 f.close()