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-21

Карпова А.С.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

**Задание по варианту**

Необходимо реализовать 9 вариант задания, а именно: 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.

**Описание реализации**

Для решения данной задачи была написана программа, которая запрашивает у пользователя размер прямоугольной матрицы, заполняет её случайными целыми числами в диапазоне от 0 до 9, а затем подсчитывает количество нулевых элементов, которые находятся в верхней части матрицы в заданном количестве строк и левой части матрицы в заданном количестве столбцов.

Функция matrix\_input() отвечает за получение размера матрицы от пользователя, а также за проверку корректности введенных значений. В случае, если введены некорректные значения (например, количество строк или столбцов равно 0), программа запрашивает у пользователя правильные значения. Затем функция использует модуль numpy.random для генерации случайных целочисленных значений в диапазоне от 0 до 9 и создает матрицу размера, указанного пользователем.

Функция zero\_search() запрашивает у пользователя количество строк и столбцов, в которых будут искаться нулевые элементы, а затем ищет все нулевые элементы на пересечении заданных строк и столбцов. Найденные нулевые элементы суммируются, и функция возвращает их количество.

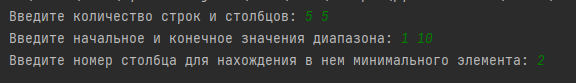
В основном блоке программы вызываются функции matrix\_input() и zero\_search() с передачей им необходимых аргументов, затем выводится сгенерированная матрица и количество нулевых элементов, которые были найдены с помощью функции zero\_search().

Общее описание реализации: программа состоит из двух функций - matrix\_input() и zero\_search() - и небольшого основного блока, в котором вызываются эти функции с передачей им необходимых аргументов и выводится результат работы программы. Каждая функция реализована как отдельный блок кода, который отвечает за конкретную задачу, и взаимодействуют между собой через параметры-аргументы. Сама программа представляет собой последовательность вызовов функций с передачей им необходимых параметров и выводом результата.

**Описание возникших затруднений**

Работа c библиотекой numpy была произведена впервые, возникли небольшие трудности из-за отсутствия русскоязычной документации, но прибегнув к возможностям интернета задача была реализована.

**Результаты работы программы**

*Рис. 1.1 Результат работы программы*

**Код программы**

Файл matrix\_processing.py:

import numpy as np  
import numpy.random as rand  
  
  
def matrix\_input():  
 try:  
 n = int(input("Введите количество строк в матрице: "))  
 m = int(input("Введите количество столбцов в матрице: "))  
  
 if n == m or n == 0 or m == 0:  
 print("Это не прямоугольная или некорректная матрица. Попробуйте еще раз.")  
 matrix\_input()  
 return rand.randint(0, 10, (n, m)), n, m  
 except ValueError:  
 print("Вы ввели некорректное значение. Попробуйте еще раз.")  
 matrix\_input()  
  
  
def zero\_search(matrix, n, m):  
 try:  
 l = int(input("Введите количество верхних строк в матрице, в которых будем искать нулевые элементы: "))  
 k = int(input("Введите количество 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. Попробуйте еще раз")  
 zero\_search(matrix, n, m)  
 zero\_counter = 0  
 for i in range(l):  
 for j in range(k):  
 if matrix[i][j] == 0:  
 zero\_counter += 1  
 return zero\_counter  
 except ValueError:  
 print("Вы ввели некорректное значение. Попробуйте еще раз.")  
 zero\_search(matrix, n, m)

Файл main.py:

from matrix\_processing import matrix\_input, zero\_search  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 matrix, n, m = matrix\_input()  
 print(f"Матрица:\n {matrix}")  
 ans = zero\_search(matrix, n, m)  
 print(f"Колчество нулевых элементов в матрице {ans}")