Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

Разработка профессиональных приложений

(название дисциплины)

**Отчёт по лабораторной работе №2**

«Двумерные структуры данных»

(название (тема) работы)

**Вариант №13**

Выполнил:

студент группы ИВТАПбд-21

Молчанов А. В.

Проверил(а):

преподаватель кафедры «ВТ»

Исхаков И. И.

Ульяновск

2023

**Постановка задачи**

Выполнить обработку элементов прямоугольной матрицы A, имеющей N строк и M столбцов. Разделить элементы каждого столбца матрицы на элемент этого столбца с наибольшим значением.

**Описание реализации**

В соответствии с требованиями к программе, она была реализована с использованием библиотеки под названием NumPy, которая позволяет производить более сложные математические вычисления, чем стандартные. В нашем случае — это создание матрицы при помощи метода eye и работа с её значениями при помощи их перебора двумя вложенными циклами for.

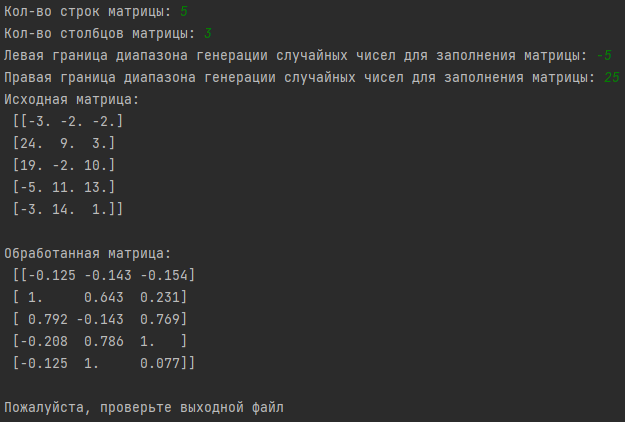
Помимо этого, реализация соответствуют прочим требованиям, предъявляемым к программе, а именно:

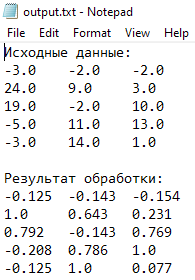
1. Наличие простейшего пользовательского интерфейса для ввода размерности матрицы и границ для автоматической генерации её случайных значений;
2. Сохранение результатов выполнения программы (исходные данные и результат обработки) в выходной файл output.txt;
3. Комментарии к исходному коду;
4. Декомпозиция программы на методы НЕ производилась, так как код вышел не слишком объёмным и без повторяющихся участков кода.

**Исходный код реализации**

import numpy as np # NumPy - библиотека для математических вычислений  
import random  
  
  
f = open('C:\\Users\\Admin\\Desktop\\output.txt', 'w') # Создаём файл в указанной директории для записи (write)  
  
N = int(input("Кол-во строк матрицы: "))  
M = int(input("Кол-во столбцов матрицы: "))  
  
l = int(input("Левая граница диапазона генерации случайных чисел для заполнения матрицы: "))  
r = int(input("Правая граница диапазона генерации случайных чисел для заполнения матрицы: "))  
  
A = np.eye(N, M) # Метод eye создаёт матрицу размером N на M, на главной диагонали которой стоят единицы  
  
f.write("Исходные данные: " + '\n') # Записываем строку в файл  
  
# Проходим всю матрицу по строкам  
for i in range(N):  
 for j in range(M):  
 # Генерируем случайное целое число в диапазоне от l до r и присваиваем его текущему элементу матрицы  
 A[i][j] = random.randint(l, r)  
 f.write(str(A[i][j]) + '\t') # Записываем этот элемент в файл  
  
 f.write('\n')  
  
maxE = l - 1 # По умолчанию присваиваем максимальному числу самое минимальное значение из возможных, чтобы он прошёл хотя бы одну проверку  
  
f.write('\n')  
  
print("Исходная матрица:\n", A)  
print()  
  
f.write("Результат обработки: " + '\n')  
  
# Проходим всю матрицу по столбцам  
for j in range(M):  
 for i in range(N):  
 if A[i][j] > maxE: # Если рассматриваемый элемент матрицы больше текущего максимального  
 maxE = A[i][j] # Перезаписываем максимальный элемент  
 else:  
 # Как только столбец закончился, начинаем идти по нему снова и делить каждый элемент на найденный максимальный в этом столбце  
 for k in range(N):  
 A[k][j] = float('{:.3f}'.format(A[k][j] / maxE))  
  
 maxE = l - 1  
  
# Переписываем всю матрицу в файл  
for i in range(N):  
 for j in range(M):  
 f.write(str(A[i][j]) + '\t')  
  
 f.write('\n')  
  
f.close()  
  
print("Обработанная матрица:\n", A)  
print()  
print("Пожалуйста, проверьте выходной файл")

**Пример работы программы**





**Описание возникших затруднений**

Основные затруднения при выполнении данной лабораторной работы были связаны с первичным знакомством с библиотекой NumPy и её методами для создания и работы с матрицами.

Также, в процессе работы возникли некоторые трудности с тем, чтобы зафиксировать строку (столбец) и идти только по элементам одного столбца (строки). Для этого внешний цикл обхода матрицы должен идти по столбцам, а внутренний — по строкам.

**Описание альтернативных способов решения**

Так как тема данной лабораторной работы — это двумерные структуры данных, то их альтернативой в языке программирования Python могут выступать двумерные массивы (списки), которые пусть и не уступают матрицам в простоте работы с ними, всё же предназначены немного для других задач (которые были разобраны в 1 лабораторной работе).

Таким образом, альтернативные способы решения у данной задачи есть, но они не имеют особого смысла по сравнению с реализацией этой же задачи при помощи матриц. Однако, стоит отметить, что если нам необходимы более, чем двумерные структуры данных, то многомерные массивы (списки) выглядят более предпочтительными по сравнению с матрицами.

**Вывод**

В ходе выполнения данной лабораторной работы были получены практические навыки работы с математической библиотекой языка Python под названием NumPy, а именно с матрицами из неё, их созданием, заполнением и обработкой.

Также, немного была затронута тема работы с файлами, которая, на первый взгляд, практически ничем не отличается от работы с ними в других языках программирования, таких как C++ и Java