

**МИНОБРНАУКИ РОССИИ**

**федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технологический университет «СТАНКИН»**

**(ФГБОУ ВО «МГТУ «СТАНКИН»)**

|  |  |
| --- | --- |
| **Институт**  **информационных**  **технологий** | **Кафедра**  **информационных систем** |

**Основная образовательная программа 09.03.02  
«Информационные системы и технологии»**

**Отчет по дисциплине «Веб-программирование»**

**по лабораторной работе №3**

|  |  |
| --- | --- |
| Студент  группы ИДБ-19-05 | Михайлуца Н.А. |
|  |  |
| Преподаватель | Кайшев Д.А. |

Ссылка на лабораторную- <https://github.com/Yyrrr/3LR_WEB> .

**ЗАДАНИЕ.**

Написать класс Matrix, который будет прообразом математического объекта(для простоты квадратная матрица). Определить для него методы сравнения на основе детерминанта этой матрицы - если детерминант первой матрицы больше, то и матрица больше. Для остальных операций аналогично. Так же реализовать операции сложения, умножения двух матриц.

Код решения представлен ниже.

import numpy # удобная библиотека для сравнения

import copy  # для копирования

class Matrix:

    def \_\_init\_\_(self, znachenie):#конструктор

        self.matrix = copy.deepcopy(znachenie)

    def \_\_str\_\_(self):# Печать

        strc = ''

        for i in range(len(self.matrix)):

            for j in range(len(self.matrix[i])):

               strc=strc+" "+ str(self.matrix[i][j])

            strc=strc+"\n"

        return strc

    def \_\_getitem\_\_(self, idx):#доступ по индексу

        return self.matrix[idx]

    def \_\_gt\_\_(self, ar):#>

        return numpy.linalg.det(self.matrix) > numpy.linalg.det(ar.matrix)

    def \_\_lt\_\_(self, ar):#<

        return numpy.linalg.det(self.matrix) < numpy.linalg.det(ar.matrix)

    def \_\_eq\_\_(self, ar):#=

        return numpy.linalg.det(self.matrix) == numpy.linalg.det(ar.matrix)

    def \_\_mul\_\_(self, ar):#\*

        res = Matrix([[0,0],[0,0]])

        for i in range(len(self.matrix)):

            for j in range(len(ar.matrix[0])):

                for k in range(len(ar.matrix)):

                    res[i][j] += self[i][k] \* ar[k][j]

        return res

    def \_\_add\_\_(self, ar):#+

        res = []

        numbers = []

        for i in range(len(self.matrix)):

            for j in range(len(self.matrix[0])):

                sum = ar[i][j] + self[i][j]

                numbers.append(sum)

                if len(numbers) == len(self.matrix):

                    res.append(numbers)

                    numbers = []

        return Matrix(res)

m1 = Matrix([[1,1],[2,2]])

m2 = Matrix([[1,2],[3,4]])

print("Первая матрица \n", m1)

print("Вторая матрица \n", m2)

print("Сумма матриц")

print(m1 + m2)

print("Произведение матриц")

print(m1 \* m2)

if m1 > m2:

    print("Первая матрица больше второй")

elif m1 < m2:

    print("Вторая матрица больше первой")

else:

    print("Матрицы равны!")

**Выводы:** в данной лабораторной работе мы научились создавать простые классы на языке python, изучили перегрузку операторов и применили полученные знания на практике.