Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Курский государственный университет»

Кафедра программного обеспечения и администрирования информационных систем

Направление подготовки математическое обеспечение и администрирование информационных систем

Форма обучения очная

**Отчет**

**по лабораторной работе №1.1**

«Метод наименьших квадратов»

дисциплина «Методы вычислений»

Выполнил:

студент группы 313.1 Козявин М.С.

Проверил:

доцент кафедры АГиТОМ Селиванова И. В.

Курск, 2023

**Цель:** Изучение особенностей применения методов аппроксимации.

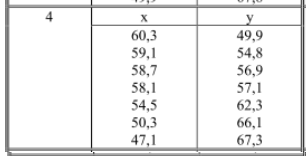
**Задание:**

1. С помощью метода наименьших квадратов вычислить значение функции в указанной точке.

2. По исходной таблице данных рассчитать параметры следующих функций: - линейной; – степенной; – показательной.

3. Построить в Excel графики функций с полученными значениями и по точкам.

**Вариант задания:**



**Код программы:**

import math

*# x\_array = [60.1, 59.2, 58.6, 55.4, 53.1, 52, 49.9]*

*# y\_array = [49, 52.1, 53.2, 56.6, 59.5, 66.6, 67.8]*

x\_array = [1, 2, 3, 4, 5, 6]

y\_array = [1.0, 1.5, 3.0, 4.5, 7.0, 8.5]

a = *float*("inf")

b = *float*("inf")

n = len(x\_array)

xs = []

ys = []

mode = 0  *# 0 - линейная, 1 - степенная, 2 - показательная*

x\_to\_solve = 2  *# число для вычисления*

*def* sum\_x():

    res = 0

    for x in xs:

        res += x

    return res

*def* sum\_y():

    res = 0

    for y in ys:

        res += y

    return res

*def* sum\_xy():

    res = 0

    for i in range(n):

        res += xs[i] \* ys[i]

    return res

*def* sum\_x\_square():

    res = 0

    for x in xs:

        res += x\*x

    return res

*def* calc():

    res\_a = (n \* sum\_xy() - sum\_x() \* sum\_y()) / (n \* sum\_x\_square() - sum\_x()\*sum\_x())

    res\_b = (1/n) \* sum\_y() - res\_a \* (1/n) \* sum\_x()

    return [res\_a, res\_b]

if mode == 0:

    xs = [\*x\_array]

    ys = [\*y\_array]

    r = calc()

    print("Линейная функция, a:", r[0], "b:", r[1])

    if x\_to\_solve < *float*("inf"):

        print("x:", x\_to\_solve, "y:", r[0]\*x\_to\_solve+r[1])

elif mode == 1:

    for i in range(n):

        xs.append(math.log(x\_array[i]))

        ys.append(math.log(y\_array[i]))

    r = calc()

    print("Cтепенная функция, a:", r[0], "b:", math.exp(r[1]))

    if x\_to\_solve < *float*("inf"):

        print("x:", x\_to\_solve, "y:", r[1]\*math.pow(x\_to\_solve, r[0]))

elif mode == 2:

    xs = [\*x\_array]

    for i in range(n):

        ys.append(math.log(y\_array[i]))

    r = calc()

    print("Показательная функция, a:", r[0], "b:", math.exp(r[1]))

    if x\_to\_solve < *float*("inf"):

        print("x:", x\_to\_solve, "y:", r[1]\*math.exp(r[0]\*x\_to\_solve))

**Тестирование:**

