ФГБОУ ВО Чувашский государственный

университет имени И.Н. Ульянова

**Лабораторная работа № 3**

**"Организация циклов"**

Вариант № 11

Выполнил студент ЭЭ-21-21

Михайлов Николай Алексеевич

Чебоксары 2022

**Цели работы:**

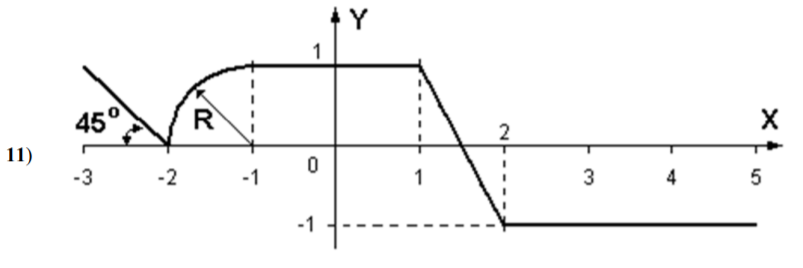
Дать студентам практический навык в использовании базовых конструкций структурного программирования - операторов цикла. Работа составлена из трёх заданий.

**Краткие теоретические сведения:**

При выполнении работы было использовано:

* Структурное программирование — парадигма программирования, в основе которой лежит представление программы в виде иерархической структуры блоков.
* Прошлые лабораторные работы
* Встроенный модуль *random*
* Углубленное изучение форматирования строк

**Рабочее задание:**

1) Вычислить и вывести на экран в виде таблицы значения функции, заданной графически (см. задание 1 лабораторной работы № 2, стр. 21), на интервале от Xнач до Xкон с шагом dx. Интервал и шаг задать таким образом, чтобы проверить все ветви программы. Таблицу снабдить заголовком и шапкой. 

**import** random

**def** **get\_value**(x:float):

    """

    Находит значение функции в точке

    Args:

        x (float): точка, в которой хотим полуить значение

    Returns:

        float: Значение функции в этой точке

    """

**if** x **<=** -2:

**return** -x - 2

**elif** -2 **<** x **<=** -1:

**return** (1 - (x + 1)\*\*2)\*\*0.5

**elif** -1 **<** x **<=** 1:

**return** 1**.**

**elif** 1 **<** x **<=** 2:

**return** -2\*x + 3

**elif** x **>** 2:

**return** -1**.**

**def** **lab3\_1**():

    """

    Печатает таблицу с значениями функции на промежутке [x\_start;x\_finish] с заданным шагом

    """

    x\_start = **float**(input('x\_start = '))

    x\_finish = **float**(input('x\_finish = '))

    step = **float**(input('step = '))

    # *-3 3 0.5*

    x = x\_start

**print**("+--------+--------+")

**print**("I   X    I    Y   I")

**print**("+--------+--------+")

**while** x **<=** x\_finish:

**print**(f"I{x: 7.2f} I{**get\_value**(x): 7.2f} I")

        x += step

**print**("+--------+--------+")

**lab3\_1**()

**Результат:**

x\_start = -6

x\_finish = 3

step = 1.25

+--------+--------+

I X I Y I

+--------+--------+

I -6.00 I 4.00 I

I -4.75 I 2.75 I

I -3.50 I 1.50 I

I -2.25 I 0.25 I

I -1.00 I 1.00 I

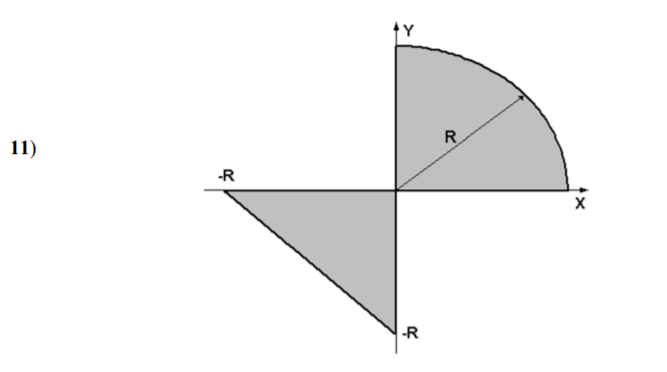
I 0.25 I 1.00 I

I 1.50 I 0.00 I

I 2.75 I -1.00 I

+--------+--------+

2) Для десяти выстрелов, координаты которых задаются генератором случайных чисел, вывести текстовые сообщения о попадании в мишень (см. лабораторная работа № 2, задание 2, стр.31).



**def** **y\_1**(x, r):

    """

    Находит значение функции в точке X, с учётом параметра r

    Args:

        x (float): точка, в которой хотим полуить значение

        r (float): параметр функции

    Returns:

        float: Значение функции в этой точке, с учётом параметра

    """

**return** -x - r

**def** **y\_2**(x, r):

    """

    Находит значение функции в точке X, с учётом параметра r

    Args:

        x (float): точка, в которой хотим полуить значение

        r (float): параметр функции

    Returns:

        float: Значение функции в этой точке, с учётом параметра

    """

**return** (r\*\*2 - x\*\*2)\*\*0.5

**def** **lab3\_2**():

    """

    Выводит таблицу с случайными коорлинатами точками и входят ли они в область, заданную функцией

    """

    r = **float**(input('Введите R = '))

    y\_0 = 0

**print**("+--------+--------+-------+")

**print**("I   X    I    Y   I  RES  +")

**print**("+--------+--------+-------+")

**for** \_ **in** **range**(20):

        x, y = random**.uniform**(-r, r), random**.uniform**(-r, r)

**if** -r **<=** x **<=** 0:

            belong = (**y\_1**(x, r) **<=** y **<=** y\_0)

**elif** 0 **<=** x **<=** r:

            belong =  (y\_0 **<=** y **<=** **y\_2**(x, r))

**else**:

            belong = **False**

**print**(f"I{x: 7.2f} I{y: 7.2f} I  {['NO ','YES'][belong]}  I")

**print**("+--------+--------+--------+")

**lab3\_2**():

**Результат:**

Введите R = 5

+--------+--------+-------+

I X I Y I RES +

+--------+--------+-------+

I 3.28 I 0.03 I YES I

I -0.14 I 4.74 I NO I

I -1.71 I -3.37 I NO I

I 4.77 I 3.61 I NO I

I 0.71 I 4.46 I YES I

I 0.25 I -1.20 I NO I

I 3.05 I 3.93 I YES I

I -2.37 I 2.61 I NO I

I -2.87 I 1.99 I NO I

I 3.99 I -2.03 I NO I

I 1.23 I -2.30 I NO I

I 1.42 I -4.89 I NO I

I 1.61 I -0.64 I NO I

I -0.57 I -0.05 I YES I

I 0.90 I -1.38 I NO I

I -0.53 I -4.78 I NO I

I 0.98 I -0.51 I NO I

I 1.06 I -3.45 I NO I

I -1.30 I -1.80 I YES I

I -2.86 I -3.81 I NO I

+--------+--------+--------+

3) Вычислить и вывести на экран в виде таблицы значения функции, заданной с помощью ряда Тейлора, на интервале от Xнач до Xкон с шагом dx с точностью **e**. Таблицу снабдить заголовком и шапкой. Каждая строка таблицы должна содержать значение аргумента, значение функции и количество просуммированных членов ряда.



**def** **mr\_taylor**(x, eps):

    """

    Возвращает вычисленное с помощью рядов Тейлора значение в данной точке с заданной точностью

    Args:

        x (float): заданная точка

        eps (float): точность вычисления

    Returns:

        float: искомое значение

        int: использованное количество слагаемых

    """

    n = 1

    Arth = x

    Arth\_add = Arth

**while** **abs**(Arth\_add) **>** eps:

        k = (1 - 2/(2\*n + 3))/x\*\*2

        Arth\_add \*= k

        Arth += Arth\_add

        n += 1

**return** Arth, n

**def** **lab3\_3**():

    x\_start = **float**(input('x\_start = '))

    x\_finish = **float**(input('x\_finish = '))

    step = **float**(input('step = '))

    eps = **float**(input('eps(погрешность) = '))

**print**("+--------+--------+-------+")

**print**("I   X    I    Y   I   N   +")

**print**("+--------+--------+-------+")

    x = x\_start

**while** x **<=** x\_finish:

        Arth, n = **mr\_taylor**(x, eps)

**print**(f"I{x: 7.2f} I{Arth: 7.2f} I{n: ^7}I")

        x += step

**print**("+--------+--------+-------+")

**lab3\_3**()

**Результат:**

x\_start = -14

x\_finish = 20

step = 4

eps(погрешность) = 3e-4

+--------+--------+-------+

I X I Y I N +

+--------+--------+-------+

I -14.00 I -14.04 I 3 I

I -10.00 I -10.06 I 4 I

I -6.00 I -6.10 I 4 I

I -2.00 I -2.37 I 7 I

I 2.00 I 2.37 I 7 I

I 6.00 I 6.10 I 4 I

I 10.00 I 10.06 I 4 I

I 14.00 I 14.04 I 3 I

I 18.00 I 18.03 I 3 I

+--------+--------+-------+

**Выводы:**

При выполнении лабораторной работы были составлены функции, что позволило сделать код компактным.

В 3 задании был совершён переход от задачи рекурсии к динамическому программированию.