

**Московский государственный технический
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»
Лабораторная работа №1

Выполнил:

Студент ИУ5-34Б

Флоринский В. А.

Подпись и дата:

Проверил:

Преподаватель каф. ИУ5

Гапанюк Ю. Е.

Подпись и дата:

Москва, 2023 г.

Задание.

Задание:

Разработать программу для решения [биквадратного уравнения](#).

1. Программа должна быть разработана в виде консольного приложения на языке Python.
2. Программа осуществляет ввод с клавиатуры коэффициентов A, B, C, вычисляет дискриминант и ДЕЙСТВИТЕЛЬНЫЕ корни уравнения (в зависимости от дискриминанта).
3. Коэффициенты A, B, C могут быть заданы в виде параметров командной строки ([вариант задания параметров приведен в конце файла с примером кода](#)). Если они не заданы, то вводятся с клавиатуры в соответствии с пунктом 2. [Описание работы с параметрами командной строки](#).
4. Если коэффициент A, B, C введен или задан в командной строке некорректно, то необходимо проигнорировать некорректное значение и вводить коэффициент повторно пока коэффициент не будет введен корректно. Корректно заданный коэффициент - это коэффициент, значение которого может быть без ошибок преобразовано в действительное число.
5. Дополнительное задание 1 (*). Разработайте две программы на языке Python - одну с применением процедурной парадигмы, а другую с применением объектно-ориентированной парадигмы.
6. Дополнительное задание 2 (*). Разработайте две программы - одну на языке Python, а другую на любом другом языке программирования (кроме C++).

Текст программ.

Данная ЛР разработана двумя путями: с помощью ООП и с помощью процедурной парадигмы. LR1OOP.py – решение задачи с помощью ООП, LR1P.py – с помощью процедурной парадигмы.

LR1OOP.py:

```
import sys

class Equation:

    def __init__(self, a = 1, b = 1, c = 1, arr = []):
        self.a = a
        self.b = b
        self.c = c
        self.arr = arr

    def get_coef(self):
        ans = 0

        try:
            arr = [float(i) for i in sys.argv[1:]]
            if (len(arr) == 3 and arr[0] != 0):
                print("Взяты параметры из cmd")
                #return arr
                self.a = arr[0]
                self.b = arr[1]
                self.c = arr[2]
                ans = 1
            else:
                print("Ошибка при взятии аргументов из cmd")
        except:
            print("Ошибка при взятии аргументов из cmd")
```

```

while ans == 0:
    try:
        print ("Введите 3 действительных числа через пробел (A, B, C)")
        arr = list(map(float, input().split())) #[float(i) for i in
input().split()]
        if (len(arr) == 3 and arr[0] != 0):
            self.a = arr[0]
            self.b = arr[1]
            self.c = arr[2]
            ans = 1
        else:
            print("Error")
            ans = 0
    except:
        print("Error")
        ans = 0

def solve(self):
    solutions = []
    d = float(self.b**2 - 4 * self.a * self.c)
    if (d == 0):
        x = -self.b / (2 * self.a)
        solutions.append(x)
    elif (d > 0):
        x1 = (-self.b + d**0.5) / (2 * self.a)
        x2 = (-self.b - d**0.5) / (2 * self.a)
        solutions.append(x1)
        solutions.append(x2)

    self.arr = solutions

def see_eq(self):
    print(f"{self.a}x^2", end="")

    if (self.b > 0): print(f"+{self.b}x", end="")
    elif (self.b < 0): print(f"{self.b}x", end="")

    if (self.c > 0): print(f"+{self.c}")
    elif (self.c < 0): print(f"{self.c}")

    print()

    if(len(self.arr) == 0): print("Нет корней")
    elif(len(self.arr) == 1): print(f"x = {self.arr[0]}")
    else: print(f"x1 = {self.arr[0]}, x2 = {self.arr[1]}")

if __name__ == "__main__":

```

```
obj = Equation()
obj.get_coef()
obj.solve()
obj.see_eq()
```

LR1P.py:

```
import sys

def get_param():

    try:
        arr = [float(i) for i in sys.argv[1:]]
        if (len(arr) == 3 and arr[0] != 0):
            print("Взяты параметры из cmd")
            return arr
        else:
            print("Ошибка при взятии аргументов из cmd")
    except:
        print("Ошибка при взятии аргументов из cmd")

    ans = 0
    while ans == 0:
        try:
            print ("Введите 3 действительных числа через пробел (A, B, C)")
            arr = list(map(float, input().split())) #[float(i) for i in
input().split()]
            if (len(arr) == 3 and arr[0] != 0):
                return arr
            else:
                print("Error")
                ans = 0
        except:
            print("Error")
            ans = 0

#arr[0, 1, 2] - a, b, c

def solve(arr):
    solutions = []
    d = float(arr[1]**2 - 4 * arr[0] * arr[2])
    if (d == 0):
        x = -arr[1] / (2 * arr[0])
        solutions.append(x)
    elif (d > 0):
        x1 = (-arr[1] + d**0.5) / (2 * arr[0])
        x2 = (-arr[1] - d**0.5) / (2 * arr[0])
        solutions.append(x1)
```

```

        solutions.append(x2)

    return solutions

def main():
    arr = get_param()
    solutions = solve(arr)
    if (len(solutions) == 2):
        print(f"X1 = {solutions[0]}; X2 = {solutions[1]}")
    elif (len(solutions) == 1):
        print(f"X = {solutions[0]}")
    else:
        print("Корней нет")

if '__main__' == __name__:
    main()

```

Примеры работы.

Запускаем LR1OOP.py через командную строку, но параметры на вход не даем:

```

C:\Users\vladi\OneDrive\Рабочий стол\LR3thSem\LR1>python LR1OOP.py
Ошибка при взятии аргументов из cmd
Введите 3 действительных числа через пробел (A, B, C)
2 10 3
2.0x^2+10.0x+3.0

X1 = -0.320550528229663, X2 = -4.6794494717703365

C:\Users\vladi\OneDrive\Рабочий стол\LR3thSem\LR1>

```

Запускаем LR1OOP.py через командную строку с параметрами на вход:

```

C:\Users\vladi\OneDrive\Рабочий стол\LR3thSem\LR1>python LR1OOP.py 2 10 3
Взяты параметры из cmd
2.0x^2+10.0x+3.0

X1 = -0.320550528229663, X2 = -4.6794494717703365

C:\Users\vladi\OneDrive\Рабочий стол\LR3thSem\LR1>

```

Запускаем LR1P.py через командную строку, но параметры на вход не даем:

```
C:\Users\vladi\OneDrive\Рабочий стол\LR3thSem\LR1>python LR1P.py
Ошибка при взятии аргументов из cmd
Введите 3 действительных числа через пробел (A, B, C)
4 7 9
Корней нет

C:\Users\vladi\OneDrive\Рабочий стол\LR3thSem\LR1>python LR1P.py
Ошибка при взятии аргументов из cmd
Введите 3 действительных числа через пробел (A, B, C)
2 10 3
X1 = -0.320550528229663; X2 = -4.6794494717703365

C:\Users\vladi\OneDrive\Рабочий стол\LR3thSem\LR1>python LR1P.py
Ошибка при взятии аргументов из cmd
Введите 3 действительных числа через пробел (A, B, C)
10 100 1000
Корней нет

C:\Users\vladi\OneDrive\Рабочий стол\LR3thSem\LR1>
```

Запускаем LR1P.py через командную строку с параметрами на вход:

```
C:\Users\vladi\OneDrive\Рабочий стол\LR3thSem\LR1>python LR1P.py 10 20 6
Взяты параметры из cmd
X1 = -0.3675444679663241; X2 = -1.632455532033676

C:\Users\vladi\OneDrive\Рабочий стол\LR3thSem\LR1>
```