

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
им. Н.Э. Баумана

Факультет “Информатика и системы управления”
Кафедра “Системы обработки информации и управления”



Дисциплина “Парадигмы и конструкции языков программирования”

Отчет по лабораторной работе №1
“Основные конструкции языка Python ”

Выполнил:
Студент группы ИУ5-34Б
Жеребенков А.Ю.
Преподаватель:
Нардид А.Н.

Москва 2025

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 А, В, С, вычисляет дискриминант и **ДЕЙСТВИТЕЛЬНЫЕ** корни уравнения (в зависимости от дискриминанта).
3. Коэффициенты А, В, С 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 А, В, С 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++).

Код(lab1)

```
import sys
import math

def get_coef(index, prompt):
    try:
        coef_str = sys.argv[index]
    except:
        print(prompt)
        coef_str = input()
    coef = float(coef_str)
    return coef

def get_roots(a, b, c):
    result = []
    D = b * b - 4 * a * c
    if D == 0.0:
        root = -b / (2.0 * a)
        result.append(root)
    elif D > 0.0:
```

```
sqD = math.sqrt(D)
root1 = (-b + sqD) / (2.0 * a)
root2 = (-b - sqD) / (2.0 * a)
result.append(root1)
result.append(root2)
return result
```

```
def main():
```

```
    a = get_coef(1, 'Введите коэффициент A:')
```

```
    b = get_coef(2, 'Введите коэффициент B:')
```

```
    c = get_coef(3, 'Введите коэффициент C:')
```

```
    roots = get_roots(a, b, c)
```

```
    finally_roots = []
```

```
    if len(roots) == 0:
```

```
        print('Нет корней')
```

```
    else:
```

```
        for el in roots:
```

```
            if el > 0:
```

```
                finally_roots.append(el**0.5)
```

```
                finally_roots.append(el**0.5 * (-1))
```

```
            elif el == 0:
```

```
                finally_roots.append(0)
```

```
    if len(finally_roots) == 1:
```

```
        print(f'Один корень: {finally_roots[0]}')
```

```
    elif len(finally_roots) == 2:
```

```
        print(f'Два корня: {finally_roots[0]} и {finally_roots[1]}')
```

```
    elif len(finally_roots) == 3:
```

```
        print(f'Три корня: {finally_roots[0]}, {finally_roots[1]}, {finally_roots[2]}')
```

```
    elif len(finally_roots) == 4:
```

```
        print(f'Четыре корня: {finally_roots[0]}, {finally_roots[1]}, {finally_roots[2]}, {finally_roots[3]}')
```

```
if __name__ == "__main__":
```

```
    main()
```

Работа приложения

Введите коэффициент A:

12

Введите коэффициент B:

13

Введите коэффициент C:

14

Нет корней

Process finished with exit code 0

```
(.venv) PS C:\Users\ADM\PycharmProjects\Zherebenkov-PkiyapLabs\lab1> python main.py 1 2 3
Нет корней
```

Введите коэффициент A:

3

Введите коэффициент B:

8

Введите коэффициент C:

-3

Два корня: 0.5773502691896257 и -0.5773502691896257