

Московский государственный технический университет им. Н.Э. Баумана

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

Курс «Базовые компоненты интернет-технологий»
Отчёт по лабораторной работе №1

Выполнил:
студент группы ИУ5-31Б
Смыслов Дмитрий
Олегович

Подпись: _____

Дата: _____

Проверил:
преподаватель каф. ИУ5
Гапанюк Юрий
Евгеньевич

Подпись: _____

Дата: _____

Москва, 2021 г.

Описание задания

1. Необходимо создать виртуальное окружение и установить в него хотя бы один внешний пакет с использованием `pip`.
2. Необходимо разработать программу, реализующую работу с классами. Программа должна быть разработана в виде консольного приложения на языке Python 3.
3. Все файлы проекта (кроме основного файла `main.py`) должны располагаться в пакете `lab_python_oop`.
4. Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета `lab_python_oop`.
5. Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры.
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры.
7. Класс «Прямоугольник» наследуется от класса «Геометрическая фигура». Класс должен содержать конструктор по параметрам «ширина», «высота» и «цвет». В конструкторе создается объект класса «Цвет фигуры» для хранения цвета. Класс должен переопределять метод, вычисляющий площадь фигуры.
8. Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус». Для вычисления площади используется константа `math.pi` из модуля `math`.
9. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны.
10. Для классов «Прямоугольник», «Квадрат», «Круг»:
 - Определите метод `repr`, который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод `format`.
 - Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
11. В корневом каталоге проекта создайте файл `main.py` для тестирования Ваших классов. Создайте следующие объекты и выведите о них информацию в консоль (N - номер Вашего варианта по списку группы):
 - Прямоугольник синего цвета шириной N и высотой N.
 - Круг зеленого цвета радиусом N.
 - Квадрат красного цвета со стороной N.

- Также вызовите один из методов внешнего пакета, установленного с использованием pip.

Текст программы

```
import sys
import math

def is_float(arg):
    try:
        float(arg)
        return True
    except ValueError:
        return False

def get_coefficient(index, prompt):
    try:
        coefficient_str = sys.argv[index]
        while not is_float(coefficient_str):
            coefficient_str = input(prompt)
    except:
        coefficient_str = input(prompt)
        while not is_float(coefficient_str):
            coefficient_str = input(prompt)
    return float(coefficient_str)

def get_quadratic_roots(a, b, c):
    result = []
    if a == 0:
        if b == 0:
            if c == 0:
                result.append('R')
            else:
                if c == 0:
                    result.append(0.0)
                else:
                    result.append(-c / b)
    else:
        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 get_biquadratic_roots(a, b, c):
    roots = []
    sub_roots = get_quadratic_roots(a, b, c)
    if len(sub_roots) != 0 and sub_roots[0] == 'R':
```

```

        roots.append('R')
    else:
        for sub_root in sub_roots:
            for root in get_quadratic_roots(1, 0, -sub_root):
                roots.append(root)
    return roots

def main():
    a = get_coefficient(1, 'Введите коэффициент A: ')
    b = get_coefficient(2, 'Введите коэффициент B: ')
    c = get_coefficient(3, 'Введите коэффициент C: ')

    roots = get_biquadratic_roots(a, b, c)

    len_roots = len(roots)
    if len_roots == 0:
        print('Нет корней')
    elif roots[0] == 'R':
        print('Корень - любое число')
    else:
        print('Корни', end=': ')
        for root in roots:
            print(root, end='; ')
        print()

if __name__ == "__main__":
    main()

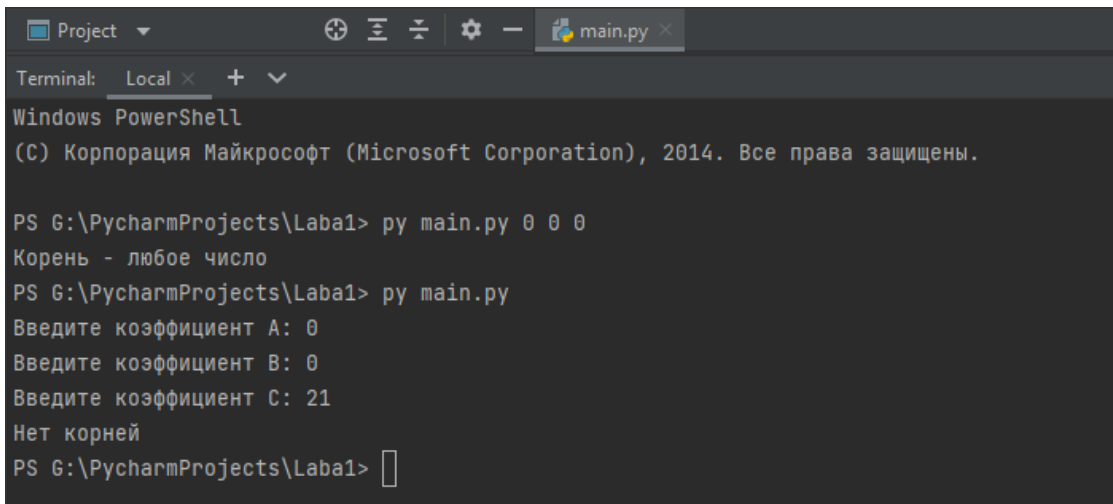
```

Экранные формы с примерами выполнения программы

Программа может обрабатывать случаи, когда у биквадратного уравнения одно, два, три или четыре решения. В случае, когда у уравнения нет корней или их бесконечно много, выводится на консоль соответствующее сообщение.

Примеры выполнения программы с параметрами, заданными в командной строке и введёнными с клавиатуры, когда корней нет или их бесконечно

МНОГО



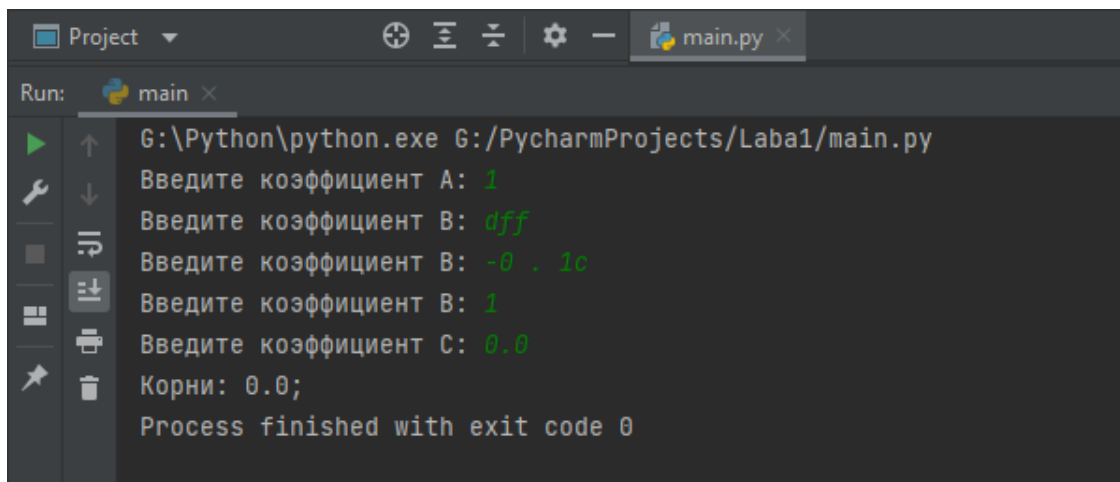
```

Project
Terminal: Local x + v
Windows PowerShell
(C) Корпорация Майкрософт (Microsoft Corporation), 2014. Все права защищены.

PS G:\PycharmProjects\Laba1> py main.py 0 0 0
Корень - любое число
PS G:\PycharmProjects\Laba1> py main.py
Введите коэффициент A: 0
Введите коэффициент B: 0
Введите коэффициент C: 21
Нет корней
PS G:\PycharmProjects\Laba1> 

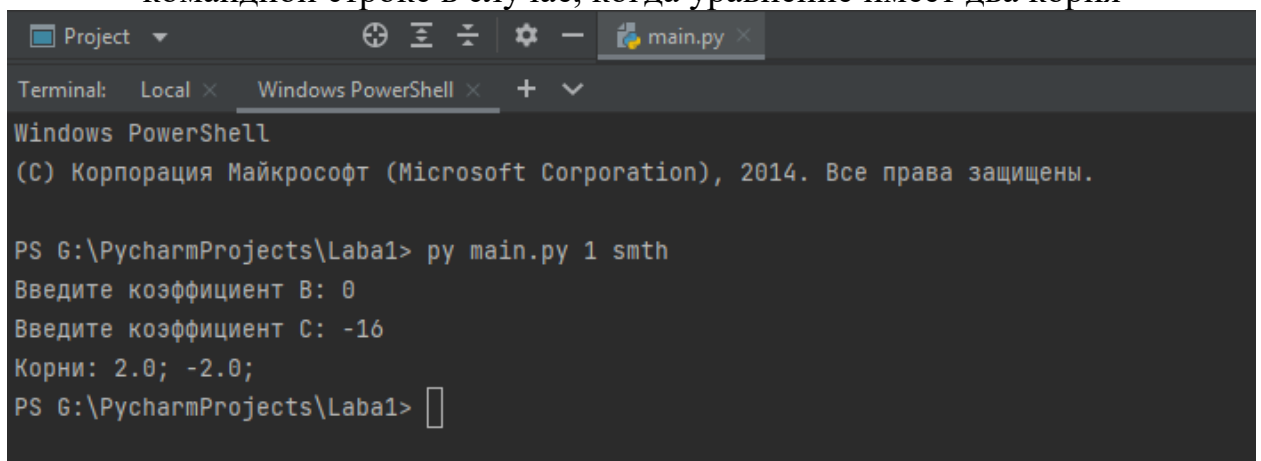
```

Пример выполнения программы при некорректном вводе коэффициентов с клавиатуры в случае, когда уравнение имеет единственный корень



```
Run: main ×
G:\Python\python.exe G:/PycharmProjects/Laba1/main.py
Введите коэффициент A: 1
Введите коэффициент B: dff
Введите коэффициент B: -0 . 1c
Введите коэффициент B: 1
Введите коэффициент C: 0.0
Корни: 0.0;
Process finished with exit code 0
```

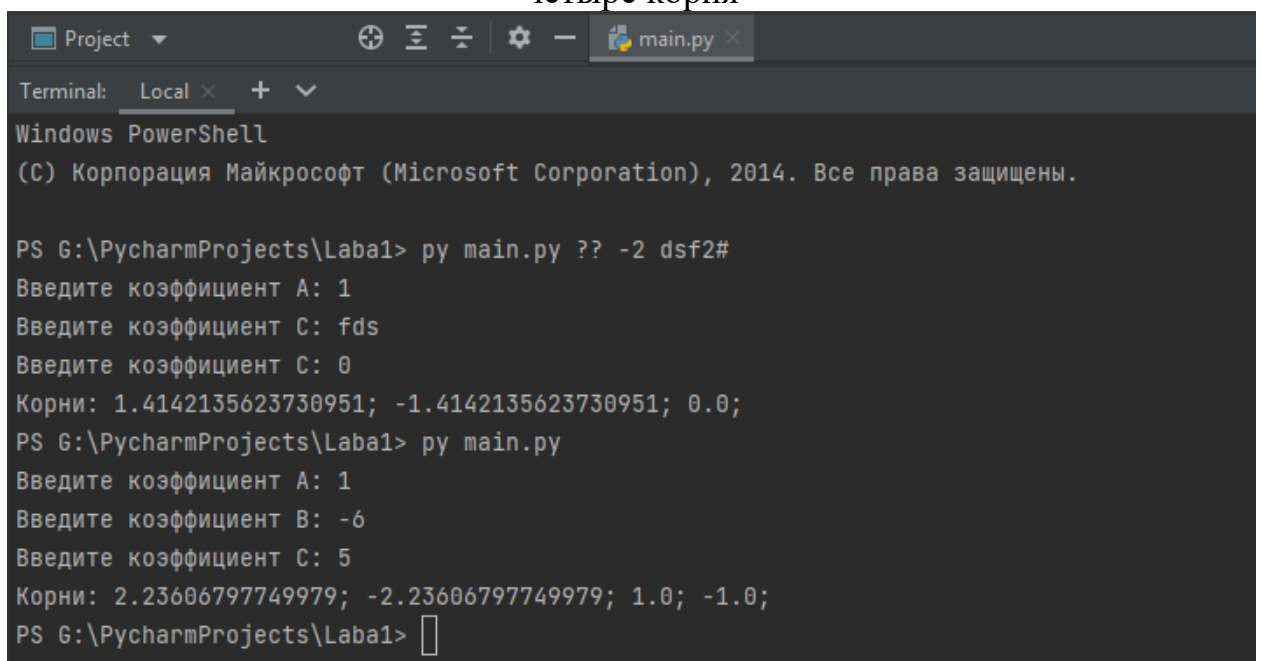
Пример выполнения программы при некорректном задании коэффициентов в командной строке в случае, когда уравнение имеет два корня



```
Terminal: Local × Windows PowerShell × + v
Windows PowerShell
(С) Корпорация Майкрософт (Microsoft Corporation), 2014. Все права защищены.

PS G:\PycharmProjects\Laba1> py main.py 1 smth
Введите коэффициент B: 0
Введите коэффициент C: -16
Корни: 2.0; -2.0;
PS G:\PycharmProjects\Laba1> 
```

Примеры выполнения программы в случае, когда уравнение имеет три или четыре корня



```
Terminal: Local × + v
Windows PowerShell
(С) Корпорация Майкрософт (Microsoft Corporation), 2014. Все права защищены.

PS G:\PycharmProjects\Laba1> py main.py ?? -2 dsf2#
Введите коэффициент A: 1
Введите коэффициент C: fds
Введите коэффициент C: 0
Корни: 1.4142135623730951; -1.4142135623730951; 0.0;
PS G:\PycharmProjects\Laba1> py main.py
Введите коэффициент A: 1
Введите коэффициент B: -6
Введите коэффициент C: 5
Корни: 2.23606797749979; -2.23606797749979; 1.0; -1.0;
PS G:\PycharmProjects\Laba1> 
```

Таким образом проверили работу программы в случае разного числа корней уравнения и при различных входных данных, как некорректных, так и правильных, заданных с клавиатуры или в консоли.