**Московский государственный технический**

**университет им. Н.Э. Баумана.**

Факультет «Информатика и системы управления»

Кафедра ИУ5. Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №6

«Основные конструкции языка Python»

|  |  |  |
| --- | --- | --- |
| Выполнил:  Макеева Е. А.  ИУ5-31Б |  | Проверил:  Нардид А. Н. |
| Подпись и дата: |  | Подпись и дата: |

Постановка задачи

Разработка алгоритма

Разработать программу для решения [биквадратного уравнения.](https://ru.wikipedia.org/wiki/%D0%A3%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5_%D1%87%D0%B5%D1%82%D0%B2%D1%91%D1%80%D1%82%D0%BE%D0%B9_%D1%81%D1%82%D0%B5%D0%BF%D0%B5%D0%BD%D0%B8#%D0%91%D0%B8%D0%BA%D0%B2%D0%B0%D0%B4%D1%80%D0%B0%D1%82%D0%BD%D0%BE%D0%B5_%D1%83%D1%80%D0%B0%D0%B2%D0%BD%D0%B5%D0%BD%D0%B8%D0%B5)

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](https://github.com/ugapanyuk/BKIT_2022/blob/main/code/lab1_code) ). Если они не заданы, то вводятся с клавиатуры в соответствии с пунктом 2. [Описание работы с параметрами командной строки.](https://realpython.com/python-command-line-arguments/#the-command-line-interface)
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.

Код программы

import sys  
import math  
  
  
class Coefficient:  
 def \_\_init\_\_(self, index, prompt):  
 self.index = index  
 self.prompt = prompt  
 self.value = self.get\_coef()  
  
 def get\_coef(self):  
 try:  
 coef\_str = sys.argv[self.index]  
 return float(coef\_str)  
 except (IndexError, ValueError):  
 print(self.prompt)  
 while True:  
 try:  
 coef\_str = input()  
 return float(coef\_str)  
 except ValueError:  
 print("Пожалуйста, введите действительное число.")  
  
  
class BiQuadraticEquation:  
 def \_\_init\_\_(self, a, b, c):  
 self.a = a  
 self.b = b  
 self.c = c  
 self.roots = []  
  
 def get\_discriminant(self):  
 return self.b \*\* 2 - 4 \* self.a \* self.c  
  
 def get\_roots(self):  
 if self.a == 0:  
 print("Уравнение не является биквадратным.")  
 return []  
  
 discriminant = self.get\_discriminant()  
 if discriminant < 0:  
 return []  
  
 y1 = (-self.b + math.sqrt(discriminant)) / (2 \* self.a)  
 y2 = (-self.b - math.sqrt(discriminant)) / (2 \* self.a)  
  
 roots = set()  
 if y1 >= 0:  
 roots.add(math.sqrt(y1))  
 roots.add(-math.sqrt(y1))  
 if y2 >= 0 and y1 != y2:  
 roots.add(math.sqrt(y2))  
 roots.add(-math.sqrt(y2))  
 self.roots = list(roots)  
  
 def display\_roots(self):  
 self.get\_roots()  
 len\_roots = len(self.roots)  
 if len\_roots == 4:  
 print(f"Уравнение имеет четыре действительных корня: {self.roots[0]}, {self.roots[1]}, {self.roots[2]} и {self.roots[3]}")  
 elif len\_roots == 3:  
 print(f"Уравнение имеет три действительных корня: {self.roots[0]}, {self.roots[1]} и {self.roots[2]}")  
 elif len\_roots == 2:  
 print(f"Уравнение имеет два действительных корня: {self.roots[0]} и {self.roots[1]}")  
 elif len\_roots == 1:  
 print(f"Уравнение имеет один действительный корень: {self.roots[0]}")  
 else:  
 print("Уравнение не имеет действительных корней.")  
  
  
def main():  
 a = Coefficient(1, 'Введите коэффициент A:').value  
 b = Coefficient(2, 'Введите коэффициент B:').value  
 c = Coefficient(3, 'Введите коэффициент C:').value  
  
 equation = BiQuadraticEquation(a, b, c)  
 equation.display\_roots()  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

**Анализ результатов**



