|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования**  **«Московский государственный технический университет имени Н.Э. Баумана**  **(национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана)** |

**Факультет «Информатика и системы управления»**

**Кафедра «Системы обработки информации и управления»**

**ОТЧЁТ ПО**

**Лабораторной работе №1**

**«Основные конструкции языка Python»**

Выполнил: Проверил:

студент группы ИУ5-35Б преподаватель каф. ИУ5

Нгуен Зуи Лам Нардид А.Н.

Подпись и дата: Подпись и дата:

Москва

2022

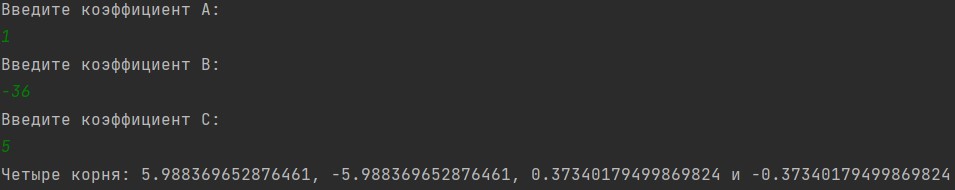
1. **Описание задания:**

Разработать программу для решения биквадратного уравнения.

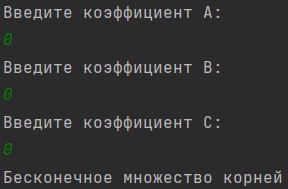
1. Программа должна быть разработана в виде консольного приложения на языке Python.
2. Программа осуществляет ввод с клавиатуры коэффициентов А, В, С, вычисляет дискриминант и ДЕЙСТВИТЕЛЬНЫЕ корни уравнения (в зависимости от дискриминанта).
3. Коэффициенты А, В, С могут быть заданы в виде параметров командной строки. 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. **Текст программы:**

import sys  
import math  
  
def get\_coef(index, prompt):  
 flag = 0  
 while flag == 0:  
 try:  
 coef\_str = sys.argv[index]  
 except:  
 print(prompt)  
 coef\_str = input()  
 try:  
 print("hhhh")  
 coef = float(coef\_str)  
 flag = 1  
 except ValueError:  
 print("lala")  
 print('Вы ввели не число! Попробуйте снова')  
 return coef  
  
def get\_roots(a, b, c):  
 result = []  
 D = b \* b - 4 \* a \* c  
 if a == 0.0 and b != 0.0:  
 if -c / b >= 0.0:  
 root = math.sqrt(-c/b)  
 if root != 0.0:  
 result.append(-root)  
 result.append(root)  
 else:  
 result.append(abs(root))  
 elif a == 0.0 and b == 0.0:  
 if c == 0.0:  
 print('Бесконечное множество корней')  
 exit(1)  
 else:  
 print('Нет корней')  
 exit(1)  
 elif D == 0.0:  
 if -b / (2.0 \* a) > 0.0:  
 root = math.sqrt(-b / (2.0 \* a))  
 if root != 0.0:  
 result.append(root)  
 result.append(-root)  
 else:  
 result.append(abs(root))  
 elif D > 0.0:  
 if (-b + math.sqrt(D)) / (2.0 \* a) > 0.0:  
 root1 = math.sqrt((-b + math.sqrt(D)) / (2.0 \* a))  
 if root1 != 0.0:  
 result.append(root1)  
 result.append(-root1)  
 else:  
 result.append(abs(root1))  
 if (-b - math.sqrt(D)) / (2.0 \* a) > 0.0:  
 root2 = math.sqrt((-b - math.sqrt(D)) / (2.0 \* a))  
 if root2 != 0.0:  
 result.append(root2)  
 result.append(-root2)  
 else:  
 result.append(abs(root2))  
 return result  
  
def main():  
 a = get\_coef(1, 'Введите коэффициент А:')  
 b = get\_coef(2, 'Введите коэффициент B:')  
 c = get\_coef(3, 'Введите коэффициент C:')  
 roots = get\_roots(a, b, c)  
 if len(roots) == 0:  
 print('Нет корней')  
 elif len(roots) == 1:  
 print('Один корень: {}'.format(roots[0]))  
 elif len(roots) == 2:  
 print('Два корня: {} и {}'.format(roots[0], roots[1]))  
 elif len(roots) == 3:  
 print('Три корня: {}, {} и {}'.format(roots[0], roots[1], roots[2]))  
 elif len(roots) == 4:  
 print('Четыре корня: {}, {}, {} и {}'.format(roots[0], roots[1], roots[2], roots[3]))  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

1. **Пример выполнения программы**

1.

2.



3.

