**Московский государственный технический**

**Университет им Н.Э.Баумана**

Факультет «Информатика и системы управление»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №1

Проверил:

Преподаватель каф. ИУ5

Нардид А. Н.

Подпись и дата:

Выполнил:

Студент группы ИУ5-34Б

Малышко А. В.

Подпись и дата:

Москва 2024 г.

**Постановка задачи**

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.
6. Дополнительное задание 2 (\*). Разработайте две программы - одну на языке Python, а другую на любом другом языке программирования (кроме С++).

**Текст программы**

import sys  
import math  
  
class SquareRoots:  
  
 def \_\_init\_\_(self):  
 *'''  
 Конструктор класса  
 '''* # Объявление коэффициентов  
 self.coef\_A = 0.0  
 self.coef\_B = 0.0  
 self.coef\_C = 0.0  
 # Количество корней  
 self.num\_roots = 0  
 # Список корней  
 self.roots\_list = []  
  
 def get\_coef(self, index, prompt):  
 *'''  
 Читаем коэффициент из командной строки или вводим с клавиатуры  
 Args:  
 index (int): Номер параметра в командной строке  
 prompt (str): Приглашение для ввода коэффицента  
 Returns:  
 float: Коэффициент квадратного уравнения  
 '''* while True:  
 try:  
 # Пробуем прочитать коэффициент из командной строки  
 coef\_str = sys.argv[index]  
 except:  
 # Вводим с клавиатуры  
 print(prompt)  
 coef\_str = input()  
 # Переводим строку в действительное число  
 try:  
 # Пытаемся преобразовать строку в число  
 coef = float(coef\_str)  
 return coef  
 except ValueError:  
 print(f"Некорректное значение: '{coef\_str}'. Пожалуйста, введите действительное число.")  
  
 def get\_coefs(self):  
 *'''  
 Чтение трех коэффициентов  
 '''* self.coef\_A = self.get\_coef(1, 'Введите коэффициент А:')  
 self.coef\_B = self.get\_coef(2, 'Введите коэффициент B:')  
 self.coef\_C = self.get\_coef(3, 'Введите коэффициент C:')  
  
 def calculate\_roots(self):  
 *'''  
 Вычисление корней квадратного уравнения  
 '''* a = self.coef\_A  
 b = self.coef\_B  
 c = self.coef\_C  
 # Вычисление дискриминанта и корней  
 D = b\*b - 4\*a\*c  
 if D == 0.0:  
 root = -b / (2.0\*a)  
 #self.num\_roots = 0  
 #self.roots\_list.append(root)  
 if(root<0):  
 self.num\_roots = 0  
 elif (root == 0):  
 self.num\_roots = 1  
 self.roots\_list.append(root)  
 elif(root>0):  
 root1=root\*\*0.5  
 root2=-(root\*\*0.5)  
 self.num\_roots=2  
 self.roots\_list.append(root1)  
 self.roots\_list.append(root2)  
  
 elif D > 0.0:  
 sqD = math.sqrt(D)  
 root1 = (-b + sqD) / (2.0\*a)  
 root2 = (-b - sqD) / (2.0\*a)  
 #self.num\_roots = 2  
 #self.roots\_list.append(root1)  
 #self.roots\_list.append(root2)  
 D1=max(root1,root2)  
 D2=min(root1,root2)  
 if(D1 < 0 and D2<0):  
 self.num\_roots=0  
 elif((D1==0 and D2<0) or (D1==0 and D2==0)):  
 self.num\_roots=1  
 self.roots\_list.append(0)  
 elif(D1>0 and D2<0):  
 self.num\_roots=2  
 root3 = D1\*\*0.5  
 root4 = -(D1 \*\* 0.5)  
 self.roots\_list.append(root3)  
 self.roots\_list.append(root4)  
 elif(D1>0 and D2==0):  
 self.num\_roots = 3  
 root3 = D1 \*\* 0.5  
 root4 = -(D1 \*\* 0.5)  
 self.roots\_list.append(root3)  
 self.roots\_list.append(root4)  
 self.roots\_list.append(0)  
 elif(D1>0 and D2>0):  
 self.num\_roots = 4  
 root3 = D1 \*\* 0.5  
 root4 = -(D1 \*\* 0.5)  
 root5 = D2 \*\* 0.5  
 root6 = -(D2 \*\* 0.5)  
 self.roots\_list.append(root3)  
 self.roots\_list.append(root4)  
 self.roots\_list.append(root5)  
 self.roots\_list.append(root6)  
  
 def print\_roots(self):  
 # Проверка отсутствия ошибок при вычислении корней  
 if self.num\_roots != len(self.roots\_list):  
 print(('Ошибка. Уравнение содержит {} действительных корней, ' +\  
 'но было вычислено {} корней.').format(self.num\_roots, len(self.roots\_list)))  
 else:  
 if self.num\_roots == 0:  
 print('Нет корней')  
 elif self.num\_roots == 1:  
 print('Один корень: {}'.format(self.roots\_list[0]))  
 elif self.num\_roots == 2:  
 print('Два корня: {} и {}'.format(self.roots\_list[0], \  
 self.roots\_list[1]))  
 elif self.num\_roots == 3:  
 print('Три корня: {} и {} и {}'.format(self.roots\_list[0], \  
 self.roots\_list[1],self.roots\_list[2]))  
 elif self.num\_roots == 4:  
 print('Четыре корня: {} и {} и {} и {}'.format(self.roots\_list[0], \  
 self.roots\_list[1], \  
 self.roots\_list[2],\  
 self.roots\_list[3]))  
  
  
def main():  
 *'''  
 Основная функция  
 '''* # Создание объекта класса  
 r = SquareRoots()  
 # Последовательный вызов необходимых методов  
 r.get\_coefs()  
 r.calculate\_roots()  
 r.print\_roots()  
  
# Если сценарий запущен из командной строки  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

**Экранные формы с примерами выполнения программы**







 