Группа: ИУ5-31Б

Студент: Болотин Ю.А.

**Текст программы**

import sys  
import math  
  
  
class QuadraticEquationSolver:  
 def \_\_init\_\_(self):  
 self.a = None  
 self.b = None  
 self.c = None  
  
 def read\_coef\_from\_input(self, name):  
 while True:  
 try:  
 coef = float(input(f"Введите коэффициент {name}: "))  
 except ValueError:  
 print("Ошибка. Введите действительное число")  
 else:  
 break  
 return coef  
  
 def read\_coef(self, index, name):  
 try:  
 coef = float(sys.argv[index])  
 except IndexError:  
 coef = self.read\_coef\_from\_input(name)  
 return coef  
  
 def get\_coefs(self):  
 self.a = self.read\_coef(1, "A")  
 self.b = self.read\_coef(2, "B")  
 self.c = self.read\_coef(3, "C")  
  
 def get\_roots(self):  
 if self.a == 0:  
 if self.b == 0:  
 return []  
 else:  
 return [-1 \* self.c / self.b, ]  
  
 result = []  
 d = self.b \*\* 2 - 4 \* self.a \* self.c  
 print(f"Дискриминант: {d}")  
  
 if d > 0:  
 d\_sqrt = math.sqrt(d)  
 root1 = (-self.b + d\_sqrt) / (2.0 \* self.a)  
 root2 = (-self.b - d\_sqrt) / (2.0 \* self.a)  
 if root1 > 0:  
 result.append(math.sqrt(root1))  
 result.append(-math.sqrt(root1))  
 elif root1 == 0:  
 result.append(root1)  
 if root2 > 0:  
 result.append(math.sqrt(root2))  
 result.append(-math.sqrt(root2))  
 elif root2 == 0:  
 result.append(math.fabs(root2))  
 elif d == 0:  
 root = -self.b / (2.0 \* self.a)  
 if root > 0:  
 result.append(math.sqrt(root))  
 result.append(-math.sqrt(root))  
 elif root == 0:  
 result.append(0)  
  
 return sorted(result)  
  
 def print\_roots(self, roots):  
 roots\_number = len(roots)  
 if roots\_number == 0:  
 print("Нет корней")  
 elif roots\_number == 1:  
 print(f"Один корень: {roots[0]}")  
 elif roots\_number == 2:  
 print(f"Два корня: {roots[0]}, {roots[1]}")  
 elif roots\_number == 3:  
 print(f"Три корня: {roots[0]}, {roots[1]}, {roots[2]}")  
 elif roots\_number == 4:  
 print(f"Четыре корня: {roots[0]}, {roots[1]}, {roots[2]}, {roots[3]}")  
  
 def solve(self):  
 self.get\_coefs()  
 roots = self.get\_roots()  
 self.print\_roots(roots)  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 solver = QuadraticEquationSolver()  
 solver.solve()

**Вывод:**

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание