



**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»

Отчет по лабораторной работе №1
«Основные конструкции языка Python»

Выполнил: студент группы ИУ5-32Б Овчинников Данила

Москва, 2021 г.

Постановка задачи:

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.
- 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.

Текст программы:

```
1  import sys
2  import math
3  import os
4
5
6  def get_coef(index, prompt):
7      '''
8      Читаем коэффициент из командной строки или вводим с клавиатуры
9      Args:
10         index (int): Номер параметра в командной строке
11         prompt (str): Приглашение для ввода коэффициента
12      Returns:
13         float: Коэффициент квадратного уравнения
14      '''
15      try:
16         # Пробуем прочесть коэффициент из командной строки
17         coef_str = sys.argv[index]
18     except:
19         # Вводим с клавиатуры
20         print(prompt)
21         coef_str = input()
22     # Переводим строку в действительное число
23     try:
24         coef = float(coef_str)
25     except ValueError:
26         print('У вас ошибка, попробуйте ввести правильные данные(число)')
27         return get_coef(index, prompt)
28     return coef
```

```

33 def get_roots(a, b, c):
34     ...
35     Вычисление корней квадратного уравнения
36     Args:
37         a (float): коэффициент A
38         b (float): коэффициент B
39         c (float): коэффициент C
40     Returns:
41         list[float]: Список корней
42     ...
43
44     result = []
45     if a == 0:
46         if b != 0:
47             root = -c / b
48             if root == 0:
49                 result.append(root)
50             elif root > 0:
51                 result.append(-round(math.sqrt(root), 2))
52                 result.append(round(math.sqrt(root), 2))
53         return result
54     D = b*b - 4*a*c
55     if D == 0.0:
56         root = -b / (2.0*a)
57         if root == 0:
58             result.append(root)
59         elif root > 0:
60             result.append(-round(math.sqrt(root), 2))
61             result.append(round(math.sqrt(root), 2))
62     elif D > 0.0:
63         sqD = math.sqrt(D)
64         root1 = (-b + sqD) / (2.0*a)
65         root2 = (-b - sqD) / (2.0*a)
66         if root1 == 0.0:
67             result.append(root1)
68         elif root1 > 0.0:
69             result.append(-round(math.sqrt(root1), 2))
70             result.append(round(math.sqrt(root1), 2))
71         if root2 == 0.0:
72             result.append(root2)
73         elif root2 > 0.0:
74             result.append(-round(math.sqrt(root2), 2))
75             result.append(round(math.sqrt(root2), 2))
76     return result

```

```

78 def main():
79     '''
80     Основная функция
81     '''
82     a = get_coef(1, 'Введите коэффициент A:')
83     b = get_coef(2, 'Введите коэффициент B:')
84     c = get_coef(3, 'Введите коэффициент C:')
85     if a == 0 and b == 0 and c == 0:
86         print('Бесконечное число корней')
87     else:
88         # Вычисление корней
89         roots = get_roots(a,b,c)
90         # Вывод корней
91         len_roots = len(roots)
92         if len_roots == 0:
93             print('Нет корней')
94         elif len_roots == 1:
95             print('Один корень: {}'.format(roots[0]))
96         elif len_roots == 2:
97             print('Два корня: {} и {}'.format(roots[0], roots[1]))
98         elif len_roots == 3:
99             print('Три корня: {}, {} и {}'.format(roots[0], roots[1], roots[2]))
100        elif len_roots == 4:
101            print('Четыре корня: {}, {}, {} и {}'.format(roots[0], roots[1], roots[2], roots[3]))
102
103
104
105
106 # Если сценарий запущен из командной строки
107 if __name__ == "__main__":
108     main()
109
110 # Пример запуска
111 # qr.py 1 0 -4
112
113 os.system("pause")

```

Тестирование:

Тест №1

```

Введите коэффициент A:
0
Введите коэффициент B:
0
Введите коэффициент C:
0
Бесконечное число корней

```

Тест №2

Введите коэффициент А:

1

Введите коэффициент В:

-5

Введите коэффициент С:

6

Четыре корня: -1.73, 1.73, -1.41 и 1.41

Тест №3

Введите коэффициент А:

1

Введите коэффициент В:

-4

Введите коэффициент С:

4

Два корня: -1.41 и 1.41

Тест №4

Введите коэффициент А:

-4

Введите коэффициент В:

16

Введите коэффициент С:

0

Три корня: -0.0, -2.0 и 2.0

Тест №5

Введите коэффициент А:

1

Введите коэффициент В:

0

Введите коэффициент С:

-16

Два корня: -2.0 и 2.0

Тест №6

Введите коэффициент А:

1

Введите коэффициент В:

0

Введите коэффициент С:

10

Нет корней

Тест №7

Введите коэффициент А:

jYj

У вас ошибка, попробуйте ввести правильные данные (число)

Введите коэффициент А:

jYj

У вас ошибка, попробуйте ввести правильные данные (число)

Введите коэффициент А:

32

Введите коэффициент В:

ghf

У вас ошибка, попробуйте ввести правильные данные (число)

Введите коэффициент В:

jkhkh

У вас ошибка, попробуйте ввести правильные данные (число)

Введите коэффициент В:

0232-302

У вас ошибка, попробуйте ввести правильные данные (число)

Введите коэффициент В:

1

Введите коэффициент С:

gfh

У вас ошибка, попробуйте ввести правильные данные (число)

Введите коэффициент С:

f

У вас ошибка, попробуйте ввести правильные данные (число)

Введите коэффициент С:

0

Один корень: 0.0