



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 «ГУИМЦ»
Кафедра ИУ5 «Системы обработки информации и управления»

Отчет по лабораторной работе №1
по дисциплине «Базовые компоненты интернет-технологий»
Тема «Основные конструкции языка Python»

Выполнил:
студент группы
ИУ5Ц-42Б Дзауров И.А.
_____, “ ____ ” _____ 2022 г.

Проверил:
преподаватель кафедры
ИУ5 - Гапанюк Ю.Е.
_____, “ ____ ” _____ 2022 г.

Москва, 2022 г.

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 A , B , C , вычисляет дискриминант и **ДЕЙСТВИТЕЛЬНЫЕ** корни уравнения (в зависимости от дискриминанта).
3. Коэффициенты A , B , C могут быть заданы в виде параметров командной строки. Если они не заданы, то вводятся с клавиатуры в соответствии с пунктом 2.
4. Если коэффициент A , B , C 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.

Листинг программы

```
# This is a sample Python script.

import sys
import math

def get_coef(index, prompt):
    """
    Читаем коэффициент из командной строки или вводим с клавиатуры
    Args:
        index (int): Номер параметра в командной строке
        prompt (str): Приглашение для ввода коэффициента
    Returns:
        float: Коэффициент квадратного уравнения
    """
    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
    while True:
        try:
            coef = float(coef_str)
        except ValueError:
            print("Неверный ввод. Попробуйте еще раз")
            # Вводим с клавиатуры
            print(prompt)
            coef_str = input()
        else:
            break

    return coef

# Определение знака
def get_sign(number):
    if number >= 0:
        return '+'
    return '-'

def get_roots(a, b, c):
    """
    Вычисление корней квадратного уравнения
    Args:
        a (float): коэффициент A
        b (float): коэффициент B
        c (float): коэффициент C
    Returns:
        list[float]: Список корней
    """
```

```

result = []
D = b * b - 4 * a * c
if D == 0.0:
    root = -b / (2.0 * a)
    result.append(root)
elif D > 0.0:
    sqD = math.sqrt(D)
    root1 = (-b + sqD) / (2.0 * a)
    root2 = (-b - sqD) / (2.0 * a)
    result.append(root1)
    result.append(root2)
return result

```

```

def get_roots_biquadratic(roots):
    """

```

Вычисление корней для биквадратного уравнения исходя из результата функции - [get_roots]

Args:

list [float]: массив корней квадратного уравнения

Returns:

list [float]: массив корней биквадратного уравнения
 ...

```

result = []

```

```

for root in roots:
    if root == 0:
        result.append(root)
    elif root > 0:
        sqRoot = math.sqrt(root)
        result.append(sqRoot)
        result.append(-sqRoot)

```

```

return result

```

```

def main():
    """

```

Основная функция

...

```

a = get_coef(1, 'Введите коэффициент - [a]:')

```

```

while a == 0.0:

```

```

    print('Коэффициент - [a] в биквадратном уравнении не может равняться нулю')

```

```

    a = get_coef(1, 'Введите коэффициент - [a]:')

```

```

b = get_coef(2, 'Введите коэффициент - [b]:')

```

```

c = get_coef(3, 'Введите коэффициент - [c]:')

```

Вычисление корней для квадратного уравнения

```

roots = get_roots(a, b, c)

```

Вычисление корней для биквадратного уравнения исходя из результата функции - [get_roots]

```

roots = get_roots_biquadratic(roots)

```

```

# Вывод корней
len_roots = len(roots)

if len_roots == 0:
    print('У уравнения {}x^4 {} {}x^2 {} {} нет корней'.format(a,
get_sign(b), abs(b), get_sign(c), abs(c)))

elif len_roots == 1:
    print('У уравнения {}x^4 {} {}x^2 {} {} один корень: {}'.format(a,
get_sign(b), abs(b), get_sign(c), abs(c), roots[0]))

elif len_roots == 2:
    print('У уравнения {}x^4 {} {}x^2 {} {} два корня: {}, {}'.format(a,
get_sign(b), abs(b), get_sign(c), abs(c), roots[0], roots[1]))

elif len_roots == 3:
    print('У уравнения {}x^4 {} {}x^2 {} {} три корня: {}, {}, {}'.format(a,
get_sign(b), abs(b), get_sign(c), abs(c), roots[0], roots[1], roots[2]))

elif len_roots == 4:
    print('У уравнения {}x^4 {} {}x^2 {} {} четыре корня: {}, {}, {},
{}'.format(a, get_sign(b), abs(b), get_sign(c), abs(c), roots[0], roots[1],
roots[2], roots[3]))

# Если сценарий запущен из командной строки
if __name__ == "__main__":
    main()

```

Экранные формы с примерами выполнения программы

Пример 1:

Введите коэффициент - [a]:

1

Введите коэффициент - [b]:

-2

Введите коэффициент - [c]:

-8

У уравнения $1.0x^4 - 2.0x^2 - 8.0$ два корня: 2.0, -2.0

Пример 2:

Введите коэффициент - [a]:

4

Введите коэффициент - [b]:

-5

Введите коэффициент - [c]:

1

У уравнения $4.0x^4 - 5.0x^2 + 1.0$ четыре корня: 1.0, -1.0, 0.5, -0.5

Пример 3:

Введите коэффициент - [a]:

1

Введите коэффициент - [b]:

-25

Введите коэффициент - [c]:

144

У уравнения $1.0x^4 - 25.0x^2 + 144.0$ четыре корня: 4.0, -4.0, 3.0, -3.0

Пример 4:

Введите коэффициент - [a]:

1

Введите коэффициент - [b]:

У

Неверный ввод. Попробуйте еще раз

Введите коэффициент - [b]:

3

Введите коэффициент - [c]:

2

У уравнения $1.0x^4 + 3.0x^2 + 2.0$ нет корней