**Московский государственный технический**

**университет им. Н.Э. Баумана.**

Факультет «Информатика и системы управления»

Кафедра ИУ5. Курс «Базовые компоненты интернет-технологий»

Отчёт по лабораторной работе № 5.

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-31Б  Успенский Д.А. |  | преподаватель каф. ИУ5  Гапанюк Ю.Е. |
|  |  |  |
| Подпись и дата: |  | Подпись и дата: |

Москва, 2022 г.

**Задание**

1. Выберите любой фрагмент кода из лабораторных работ 1 или 2 или 3-4.
2. Модифицируйте код таким образом, чтобы он был пригоден для модульного тестирования.
3. Разработайте модульные тесты. В модульных тестах необходимо применить следующие технологии:
   1. TDD - фреймворк (не менее 3 тестов).
   2. BDD - фреймворк (не менее 3 тестов).
   3. Создание Mock-объектов (необязательное дополнительное задание).

Текст программы

# Файл lab01.py

import sys

import cmath

# Получение коэффициентов от пользователя

def get\_coef(index, str):

# попытка считать коэф. из командной строки

try:

coef\_str = sys.argv[index]

# ввод коэф. с клаавиатуры

except:

print(str)

coef\_str = input()

while True:

try:

# перевод стринга в действительное число

float(coef\_str)

break

except:

coef\_str = input()

coef = float(coef\_str)

return coef

def get\_roots(a,b,c):

answr = []

#расчет дискрименанта

D = b\*b - 4\*(a\*c)

#расчет корней

if D == 0:

root = -b/(2.0\*a)

answr.append(root)

elif D > 0.0:

sqD = cmath.sqrt(D)

root1 = (-b + sqD)/(2.0\*a)

root2 = (-b - sqD)/(2.0\*a)

answr.append(root1)

answr.append(root2)

return answr

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)

len\_roots = len(roots)

#вывод корней

if len\_roots == 0:

print('Нет Корней.')

if len\_roots == 1:

print('Единственный корень уравнения: {}'.format(roots[0]))

if len\_roots == 2:

print('Два корня уравнения: {} и {}'.format(roots[0],roots[1]))

#для запуска кода из командной строки

if \_\_name\_\_ == "\_\_main\_\_":

main()

# Файл test.py

import unittest

from lab1 import get\_roots

class SquareEqSolverTestCase(unittest.TestCase):

def test\_no\_root(self):

res = get\_roots(1, 11, 10)

self.assertEqual(len(res), 0)

def test\_single\_root(self):

res = get\_roots(10, 0, 0)

self.assertEqual(len(res), 1)

self.assertEqual(res, [0])

def test\_two\_roots(self):

res = get\_roots(1, -2, -8)

self.assertEqual(len(res), 2)

self.assertEqual(res, [-2, 2])

def test\_three\_roots(self):

res = get\_roots(-4, 16, 0)

self.assertEqual(len(res), 3)

self.assertEqual(res, [-2, 0, 2])

def test\_four\_roots(self):

res = get\_roots(1, -10, 9)

self.assertEqual(len(res), 4)

self.assertEqual(res, [-3, -1, 1, 3])

# Файл features/ tests2.feature

Feature: testing roots

Scenario Outline: multiple roots roots

Given <a> coef a, coef b is <b> and c is <c>

When starting function

Then we shoud see <result>

Examples:

| a | b | c | result |

| 1 | 10 | 11 | "[]" |

| 10 | 0 | 0 | "[0]" |

| 1 | -2 | -8 | "[-2, 2]" |

| 4 | 16 | 0 | "[-2, 0, 2]" |

| 1 | -10 | 9 | "[-3, 1, 1, 3]"|

Анализ Результатов

