

Факультет РТ Радиотехнический

Кафедра ИУ5 Системы обработки информации и управления

**Отчет по лабораторной работе № 4 по курсу  
Базовые компоненты интернет-технологий**

Исполнитель

Студент группы РТ5-31Б \_\_\_\_\_ Яковенко Ю.С.

“ \_\_\_\_ ” \_\_\_\_\_ 2021 г.

Проверил

Доцент кафедры ИУ5 \_\_\_\_\_ Гапанюк Ю.Е.

“ \_\_\_\_ ” \_\_\_\_\_ 2021 г.

## **Содержание**

1. Описание задания.....	3
2. Текст программы.....	3
3. Экранные формы с примерами выполнения программы .....	5

## 1. Описание задания

1. Написать тесты для своей программы решения биквадратного уравнения. Доработать программу решения биквадратного уравнения, чтобы она была пригодна для модульного тестирования.
2. В модульных тестах необходимо применить следующие технологии:  
TDD - фреймворк.  
BDD - фреймворк.  
Создание Mock-объектов.

## 3. Текст программы

### TDD.py

```
import unittest
from lab1 import get_roots

class RootsTest(unittest.TestCase):
    def test_roots(self):
        self.assertEqual(get_roots(4, -5, 1), [1, -1, 0.5, -0.5])
        self.assertEqual(get_roots(1, -2, -8), [2, -2])
        self.assertEqual(get_roots(1, 11, 10), [])

if __name__ == "__main__":
    unittest.main()
```

### BDD.feature

Feature: Test BDD

Scenario: Get roots of a biquadratic equation

Given I give coefficients 4, -5, 1

Then I get 1, -1, 0.5, -0.5 roots

Given I give coefficients 1, -2, -8

Then I get 2, -2 roots

## **test\_BDD.py**

```
import re
from pytest_bdd import scenario, given, then, parsers
from lab1 import get_roots

@scenario("BDD.feature", "Get roots of a biquadratic equation")
def test_roots():
    pass

@given(parsers.parse('I give coefficients {a:d}, {b:d}, {c:d}'), target_fixture='res')
def give_roots(a, b, c):
    return get_roots(a, b, c)

@then(parsers.parse('I get {roots} roots'))
def get_result(res, roots):
    roots = list(map(float, roots.split(", ")))
    assert res == roots
```

## **Mock.py**

```
import unittest
from unittest.mock import Mock
from lab1 import get_roots

class RootsTest(unittest.TestCase):
    def test_roots(self):
        mock = Mock(return_value=4)
        self.assertEqual(get_roots(mock(), -5, 1), [1, -1, 0.5, -0.5])

if __name__ == "__main__":
    unittest.main()
```

## 4. Экранные формы с примерами выполнения программы

### TDD

```
-----  
Ran 1 test in 0.000s
```

```
OK
```

### BDD

```
----- test session starts -----  
platform win32 -- Python 3.6.8, pytest-6.2.5, py-1.11.0, pluggy-1.0.0  
rootdir: D:\Univer\Prog\Python\lab4\venv  
plugins: bdd-5.0.0  
collected 1 item  
  
test_BDD.py . [100%]  
  
===== 1 passed in 0.25s =====
```

### Mock

```
-----  
Ran 1 test in 0.000s
```

```
OK
```