

**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 №5
«Модульное тестирование Python»

Выполнил:
студент группы ИУ5-34Б:
Козак А.А.
Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю.Е.,
Подпись и дата:

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:
 - TDD - фреймворк (не менее 3 тестов).
 - BDD - фреймворк (не менее 3 тестов).

tdd_test.py

```
import unittest
from unique import Unique

class TestUnique(unittest.TestCase):
    def test_empty(self):
        self.assertEqual(list(Unique([])), [])
    def test_ign_case_false(self):
        arr = ['a', 'A', 'a', 'b', 'B', 'B', 'B']
        self.assertEqual(list(Unique(arr)), ['a', 'A', 'b', 'B'])
    def test_ign_case_true(self):
        arr = ['a', 'A', 'a', 'b', 'B', 'B', 'B']
        self.assertEqual(list(Unique(arr, ignore_case=True)), ['a', 'b'])
    def test_with_not_str(self):
        arr = [[], 1, 2, 1, 1, [], [1], 'a', 'b', 'A', 'b', 'b', [], []]
        self.assertEqual(list(Unique(arr, ignore_case=True)), [[], 1, 2, [1], 'a', 'b', []])

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

bdd_sort.feature

```
Feature: Sort
  A function that sorts an array modulo in reverse order

  Scenario: Empty array
    Given Empty array
    When We call a function from it
    Then It should still be empty

  Scenario: Small array
    Given Small array
    When We call a function from small array
    Then It should return new array which is sorted modulo in reverse order

  Scenario: Big array
    Given Big array
    When We call a function from big array
    Then It should return new big array which is sorted modulo in reverse order
```

bdd_sort_test.py

```
1 from pytest_bdd import scenario, given, when, then
2 from sort import sort_without_lambda
3 import random as rd
4
5 @scenario('bdd_sort.feature', 'Empty array')
6 def test_empty_array():
7     pass
8
9 @given('Empty array', target_fixture="arr")
10 def give_empty_arr():
11     return []
12
13 @when("We call a function from it", target_fixture='sort_arr')
14 def call_func_empty(arr):
15     return sort_without_lambda(arr)
16
17 @then('It should still be empty')
18 def eq_empty(sort_arr):
19     assert ([] == sort_arr)
20
21
```

```

21
22 @scenario('bdd_sort.feature','Small array')
23 def test_small_array():
24     pass
25
26 @given('Small array',target_fixture= "arr")
27 def give_small_arr():
28     arr = list(range(10))
29     rd.shuffle(arr)
30     return arr
31
32 @when("We call a function from small array",target_fixture='sort_arr')
33 def call_func_small(arr):
34     return sort_without_lambda(arr)
35
36 @then('It should return new array which is sorted modulo in reverse order')
37 def eq_small(sort_arr):
38     assert (list(range(10)) == sort_arr[::-1])
39
40
41 @scenario('bdd_sort.feature','Big array')
42 def test_big_array():
43     pass
44
45 @given('Big array',target_fixture= "arr")
46 def give_big_arr():
47     return [790, 339, 313, -441, 773, 246, 737, 818, 248, -412, -514,
48             -361, -973, -686, 763, 777, -827, -5, -37, -23, 553, -92,
49             145, -620, 851, -429, 564, 986, 26, 464, -505, -366, 519, -428,
50             -676, -776, -493, -784, -377, 336, 786, -7, -243, -699, -825, -893,
51             -291, 76, -14, -535]
52
53 @when("We call a function from big array",target_fixture='sort_arr')
54 def call_func_big(arr):
55     return sort_without_lambda(arr)
56
57 @then('It should return new big array which is sorted modulo in reverse order')
58 def eq_big(sort_arr):
59     ans = [986, -973, -893, 851, -827, -825, 818, 790, 786, -784, 777,
60            -776, 773, 763, 737, -699, -686, -676, -620, 564, 553, -535, 519, -514,
61            -505, -493, 464, -441, -429, -428, -412, -377, -366, -361, 339, 336, 313,
62            -291, 248, 246, -243, 145, -92, 76, -37, 26, -23, -14, -7, -5]
63     assert (ans == sort_arr)
64
65
66

```

Тестирование

```
C:\Users\Vegard\Pictures\Lab 3>pytest tdd_test.py
===== test session starts =====
platform win32 -- Python 3.8.5, pytest-7.2.0, pluggy-1.0.0
rootdir: C:\Users\Vegard\Pictures\Lab 3
plugins: bdd-6.0.1
collected 4 items

tdd_test.py .... [100%]

===== 4 passed in 0.12s =====

C:\Users\Vegard\Pictures\Lab 3>pytest bdd_sort_test.py
===== test session starts =====
platform win32 -- Python 3.8.5, pytest-7.2.0, pluggy-1.0.0
rootdir: C:\Users\Vegard\Pictures\Lab 3
plugins: bdd-6.0.1
collected 3 items

bdd_sort_test.py ... [100%]

===== 3 passed in 0.11s =====
```