



**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 «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»**

**Домашнее задание
по дисциплине «Разработка комплексного приложения на языке Python»
по предмету Базовые компоненты интернет технологий**

**Выполнила:
студентка группы ИУ5-33Б
Валова С. В.**

**Проверил:
Преподаватель кафедры ИУ-5
Гапанюк Ю. Е.**

2022 г.

Условие задания:

1. С использованием механизма итераторов или генераторов реализуйте с помощью концепции ленивых вычислений одну из последовательностей OEIS. Примером могут являться числа Фибоначчи.
2. Для реализованной последовательности разработайте 3-5 модульных тестов, которые, в том числе, проверяют то, что последовательность поддерживает ленивые вычисления.
3. Разработайте веб-сервис с использованием фреймворка Flask, который возвращает N элементов последовательности (параметр N передается в запросе к сервису).
4. Создайте Jupyter-notebook, который реализует обращение к веб-сервису с использованием библиотеки requests и визуализацию полученных от веб-сервиса данных с использованием библиотеки matplotlib.

Текст программы

Файл app.py

```
from flask import Flask
from fibonacci import func_fib

app = Flask(__name__)

@app.route('/')
def hello():
    return 'There are the Fibonacci numbers:'

@app.route('/<int:NumberOfNumbers>')
def num(NumberOfNumbers):
    b = func_fib()
    fib_numbers = [next(b) for i in range(NumberOfNumbers)]
    return fib_numbers

if __name__ == "__main__":
    app.run(debug=True)
```

Файл fibonacci.py

```
def func_fib(n=25):
    fib1, fib2 = 0, 1
    for i in range(n):
        yield fib1
        fib1, fib2 = fib2, fib1 + fib2
```

Файл test.py

```
import unittest
from fibonacci import func_fib

class test_fib(unittest.TestCase):
    def test_1(self):
```

```

        b = func_fib()
        fib_numbers = [next(b) for i in range(5)]
        self.assertEqual(fib_numbers, [0, 1, 1, 2, 3])

    def test_2(self):
        b = func_fib()
        fib_numbers = [next(b) for i in range(2)]
        self.assertEqual(fib_numbers, [0, 1])

    def test_3(self):
        b = func_fib()
        fib_numbers = [next(b) for i in range(10)]
        self.assertEqual(fib_numbers, [0, 1, 1, 2, 3, 5, 8, 13, 21, 34])

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

```

Результат выполнения тестов:

```





PS C:\Users\svalo\PycharmProjects\HW> python tests.py
...
-----
Ran 3 tests in 0.001s

OK

```





Результаты выполнения:

← → ↻ ⓘ 127.0.0.1:5000

 Gmail  YouTube  Яндекс 

There are the Fibonacci numbers:

← → ↻ ⓘ 127.0.0.1:5000/5

 Gmail  YouTube  Яндекс 

```

[
  0,
  1,
  1,
  2,
  3
]

```

```
In [1]: import requests
import json
```

Получим данные

```
In [4]: r = requests.get('http://127.0.0.1:5000/10')
fib_numbers = list(r.json())
print(fib_numbers)
```

```
[0, 1, 1, 2, 3, 5, 8, 13, 21, 34]
```

```
In [5]: import matplotlib.pyplot as plt
```

```
In [6]: plt.plot(fib_numbers)
plt.show()
```

