Выполнил: Щепетов Дмитрий

Группа: ИУ5-32Б

**Домашнее задание по дисциплине «Базовые компоненты интернет-технологий»**

**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 test\_fibonachi.py**

import unittest

import time

def fibonachi(n):

    fib1, fib2 = 0, 1

    for i in range(n):

        fib1, fib2 = fib2, fib1 + fib2

        yield fib1

class TestEquation(unittest.TestCase):

    def test\_fibonachi\_1(self):

        exp\_res = [0, 1, 1, 2, 3, 5, 8, 13, 21, 34]

        res = [i for i in fibonachi(10)]

        print(res)

        res1 = [0] \* (len(res))

        for i in range(1,len(res)):

            res1[i] = res[i-1]

        self.assertEqual(res1, exp\_res)

    def test\_fibonachi\_2(self):

        expected\_res = []

        res = [i for i in fibonachi(0)]

        self.assertEqual(res, expected\_res)

    def test\_fibonachi\_4(self): # ленивые вычисления

        start\_time = time.time()

        s = fibonachi(999999)

        end\_time = time.time() - start\_time

        self.assertLess(end\_time, 1)

if \_\_name\_\_ == '\_\_main\_\_':

    print(fibonachi(10))

    unittest.main()

**Файл flaskp.py**

import unittest

import requests

from test\_fibonachi import fibonachi

from flask import Flask

app = Flask(\_\_name\_\_)

@app.route('/')

def index():

    t = [1,2,3]

    print(\*t, sep='\n')

    a = list(map(int,fibonachi(10)))

    return a

@app.route('/testik222')

def fib\_1():

    cnt = 17

    return f'<h1>{str(list(fibonachi(cnt)))[1:-1]}<h1>'

@app.route('/testik321')

def number(n):

    fibgen = fibonachi(n)

    res = [next(fibgen) for i in range(n)]

    return res

@app.errorhandler(404)

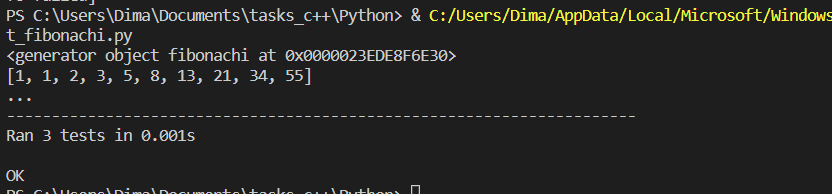
def not\_found\_error(error):

    return "Введите целое число, а не непонятно что"

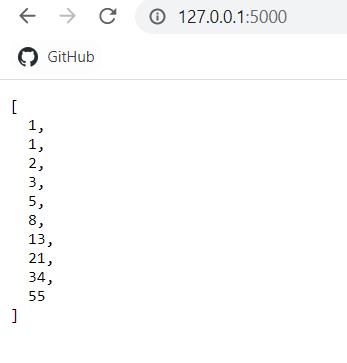
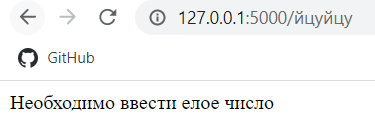
if \_\_name\_\_ == "\_\_main\_\_":

    app.run(debug = True)

**Результаты выполнения**



**Flask:**

**** ****

**Jupyter:** 