

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_fibonacci_4(self): # ленивые вычисления
        start_time = time.time()
        s = fibonacci(999999)
        end_time = time.time() - start_time
        self.assertLess(end_time, 1)

if __name__ == '__main__':
    print(fibonacci(10))
    unittest.main()

```

Файл flaskr.py

```

import unittest
import requests
from test_fibonacci import fibonacci
from flask import Flask

app = Flask(__name__)

@app.route('/')
def index():
    t = [1,2,3]
    print(*t, sep='\n')
    a = list(map(int, fibonacci(10)))
    return a

@app.route('/testik222')
def fib_1():
    cnt = 17
    return f'<h1>{str(list(fibonacci(cnt)))[1:-1]}<h1>'

@app.route('/testik321')
def number(n):
    fibgen = fibonacci(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

```
PS C:\Users\Dima\Documents\tasks_c++\Python> & C:/Users/Dima/AppData/Local/Microsoft/Windows/PowerShell/Scripts/t_fibonachi.py
<generator object fibonachi at 0x0000023EDE8F6E30>
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55]
...
-----
Ran 3 tests in 0.001s

OK
PS C:\Users\Dima\Documents\tasks_c++\Python>
```

Flask:



127.0.0.1:5000



```
[
  1,
  1,
  2,
  3,
  5,
  8,
  13,
  21,
  34,
  55
]
```



127.0.0.1:5000/йцуйцу



Необходимо ввести елое число

Jupyter:

```
Ввод [22]: import requests
import matplotlib.pyplot as plt
url = "http://127.0.0.1:5000/"
n = 10

r = requests.get(url, timeout=(10, 10))
try:
    data = r.json()
except:
    data = r.text
data
```

Out[22]: [1, 1, 2, 3, 5, 8, 13, 21, 34, 55]

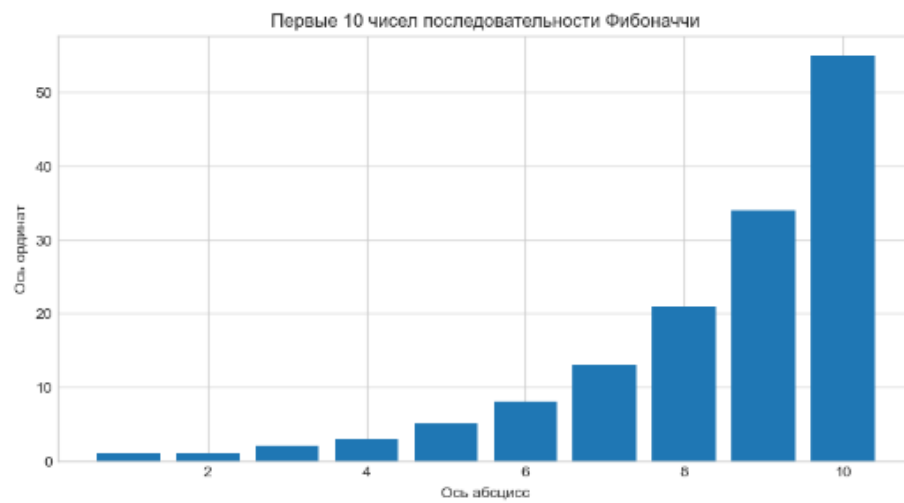
```
Ввод [15]: ! pip install requests
! pip install matplotlib
```

```
Requirement already satisfied: requests in c:\users\dima\anaconda3\lib\site-packages (2.28.1)
Requirement already satisfied: charset-normalizer<3,>=2 in c:\users\dima\anaconda3\lib\site-packages (from requests) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\dima\anaconda3\lib\site-packages (from requests) (3.3)
Requirement already satisfied: certifi=2017.4.17 in c:\users\dima\anaconda3\lib\site-packages (from requests) (2022.9.14)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\dima\anaconda3\lib\site-packages (from requests) (1.26.11)
Requirement already satisfied: matplotlib in c:\users\dima\anaconda3\lib\site-packages (3.5.2)
Requirement already satisfied: cycler>=0.10 in c:\users\dima\anaconda3\lib\site-packages (from matplotlib) (0.11.0)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\dima\anaconda3\lib\site-packages (from matplotlib) (3.0.9)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\dima\anaconda3\lib\site-packages (from matplotlib) (1.4.2)
Requirement already satisfied: numpy>=1.17 in c:\users\dima\anaconda3\lib\site-packages (from matplotlib) (1.21.5)
Requirement already satisfied: packaging>=20.0 in c:\users\dima\anaconda3\lib\site-packages (from matplotlib) (21.3)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\dima\anaconda3\lib\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: pillow>=6.2.0 in c:\users\dima\anaconda3\lib\site-packages (from matplotlib) (9.2.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\dima\anaconda3\lib\site-packages (from matplotlib) (4.25.0)
Requirement already satisfied: six>=1.5 in c:\users\dima\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.16.0)
```

```
Ввод [23]: y = data
x = list(range(1, len(y)+1))
print(y)
print(x)
```

```
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55]
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

```
Ввод [24]: fig = plt.figure(figsize = (10, 5))
plt.bar(x, y)
plt.xlabel('Ось абсцисс')
plt.ylabel('Ось ординат')
plt.title('Первые {} чисел последовательности Фибоначчи'.format(len(y)))
plt.show()
```



```
Ввод [25]: fig = plt.figure(figsize = (10, 5))  
plt.plot(x, y)  
plt.show()
```

