**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 «Базовые компоненты интернет-технологий»

Выполнил:

студент группы ИУ5-33Б

Рассказов Н.Д.

Проверил:

Гапанюк Ю. Е.

2022 г.

1. **Описание задания.**
2. С использованием механизма итераторов или генераторов реализуйте с помощью концепции ленивых вычислений [одну из последовательностей OEIS.](https://ru.wikipedia.org/wiki/%D0%9E%D0%BD%D0%BB%D0%B0%D0%B9%D0%BD-%D1%8D%D0%BD%D1%86%D0%B8%D0%BA%D0%BB%D0%BE%D0%BF%D0%B5%D0%B4%D0%B8%D1%8F_%D1%86%D0%B5%D0%BB%D0%BE%D1%87%D0%B8%D1%81%D0%BB%D0%B5%D0%BD%D0%BD%D1%8B%D1%85_%D0%BF%D0%BE%D1%81%D0%BB%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE%D1%81%D1%82%D0%B5%D0%B9) Примером могут являться [числа Фибоначчи.](https://ru.wikipedia.org/wiki/%D0%A7%D0%B8%D1%81%D0%BB%D0%B0_%D0%A4%D0%B8%D0%B1%D0%BE%D0%BD%D0%B0%D1%87%D1%87%D0%B8)
3. Для реализованной последовательности разработайте 3-5 модульных тестов, которые, в том числе, проверяют то, что последовательность поддерживает ленивые вычисления.
4. Разработайте веб-сервис с использованием фреймворка Flask, который возвращает N элементов последовательности (параметр N передается в запросе к сервису).
5. Создайте Jupyter-notebook, который реализует обращение к веб-сервису с использованием библиотеки [requests](https://requests.readthedocs.io/en/latest/) и визуализацию полученных от веб-сервиса данных с использованием библиотеки [matplotlib.](https://matplotlib.org/)
6. Текст программы.

fibonacci.py

*import* types

*import* pytest

def fibonacci\_numbers(*n*: int):

*if* (*n* <= 0):

*raise* ValueError(

            "The number of Fibonacci numbers must be greater than 0")

    a = 0

    b = 1

*for* \_ *in* range(0, *n*):

*yield* a

        a, b = b, a + b

class TestFibonacci:

    def test\_numbers(*self*):

        a = [i *for* i *in* fibonacci\_numbers(15)]

        b = [0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377]

*assert* a == b

    def test\_error(*self*):

        a = fibonacci\_numbers(-1)

*with* pytest.raises(ValueError):

*for* i *in* a:

                i += 1

    def test\_generator(*self*):

        a = fibonacci\_numbers(3)

*assert* isinstance(a, types.GeneratorType)

request.py

*import* requests

def request(*n*: int):

    url = "http://127.0.0.1:5000/"

    res = url + '/num/' + str(*n*)

    r = requests.get(res, *timeout*=(10, 10))

*try*:

        data = r.json()

*except*:

        data = r.text

*return* data

def main():

    n = input("Enter the number of numbers: ")

    print(request(n))

    n = input("Enter the number of numbers: ")

    print(request(n))

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

    main()

run.py

*'''*

*Для запуска сервиса используем команду командной строки:*

*flask --app run.py run*

*'''*

*from* flask *import* Flask

*from* fibonacci *import* fibonacci\_numbers

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

@app.route("/")

def hello\_world():

*return* "<p>Returning the Fibonacci numbers!</p>"

@app.route('/num/<int:cnt>')

def get\_fibonacci(*cnt*):

    fib\_gen = fibonacci\_numbers(*cnt*)

    res = [i *for* i *in* fib\_gen]

*return* res

1. Экранные формы с примерами выполнения программы.

