

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ
ФЕДЕРАЦИИ**
**Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Кафедра инфокоммуникаций

«Основы ветвления Git»

Отчет по лабораторной работе № 2.7

по дисциплине «Основы программной инженерии»

Выполнил студент группы ПИЖ-б-о-21-1

Образцова М. Д. .«6» декабря 2022г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил Воронкин Р.А. _____

(подпись)

Ставрополь 2022

МЕТОДИКА И ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Создала репозиторий в GitHub «OPI_LR_10» в который добавила .gitignore для работы с IDE PyCharm с ЯП Python, выбрала лицензию MIT.

Create a new repository

A repository contains all project files, including the revision history. Already have a project [Import a repository](#).

Owner *

Repository name *

obrMaria ▾

/

OPI_LR_10 ✓

Great repository names are short and memorable. Need inspiration? How about [automati](#)

Description (optional)

☒ Public

Anyone on the internet can see this repository. You choose who can commit.

☐ Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☒ Add a README file

This is where you can write a long description for your project. [Learn more](#).

Add .gitignore

Choose which files not to track from a list of templates. [Learn more](#).

.gitignore template: Python ▾

Choose a license

A license tells others what they can and can't do with your code. [Learn more](#).

License: MIT License ▾

This will set `main` as the default branch. Change the default name in your [settings](#).

ⓘ You are creating a public repository in your personal account.

Create repository

Рисунок 1 – Создание репозитория

```

M@DESKTOP-UVM9NOL MINGW64 ~/desktop/OPI_LR_10 (main)
$ git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Users/M/desktop/OPI_LR_10/.git/hooks]

M@DESKTOP-UVM9NOL MINGW64 ~/desktop/OPI_LR_10 (develop)
$

```

Рисунок 2 – Организация репозитория в соответствии с моделью ветвления git-flow

Рисунок 2 – Изменение .gitignore

```

1 x
C:\Users\vika1\AppData\Local\Microsoft\Wind
x = {'k', 'j', 'd', 'e', 'o'}
y = {'v', 'y', 'c', 'g', 'o', 'f', 'h'}
5

```

Рисунок 4 – Результат выполнения программы для 1 примера

- (1) Решите задачу: подсчитайте количество гласных в строке, введенной с клавиатуры с использованием множеств.
- (2) Решите задачу: определите общие символы в двух строках, введенных с клавиатуры.

```
1 x
C:\Users\vika1\AppData\Local\Microsoft\W
Введите строку: abcdee
Количество гласных в предложении = 3

Введите строку
abc de ff
Введите строку
ab f
общие символы = {'b', 'f', 'a'}
количесвто общих символов = 3
```

Рисунок 5 – Результат выполнения программы

Индивидуальное задание

$$\begin{aligned} A = \{c, g, h, k, y\}; \quad B = \{a, b, k, n, u\}; \quad C = \{i, j, o, y, z\}; \quad D = \{a, b, f, g, y, z\}; \\ X = (A \cup B) \cap D; \quad Y = (\bar{A} \cap D) \cup (\bar{C} / \bar{B}). \end{aligned} \quad (21)$$

```
2 x
C:\Users\vika1\AppData\Local\Microsoft\Win
x = {'a', 'g', 'y', 'b'}
y = {'a', 'z', 'f', 'b', 'k', 'n', 'u'}
```

Рисунок 7 – Результат выполнения индивидуального задания

Вопросы для защиты работы

1. Что такое множества в языке Python?

Это неупорядоченная совокупность уникальных элементов.

2. Как осуществляется создание множества в Python?

С помощью фигурных скобок. Пример: `a = {a, b, c, d}`

Существует и другой способ создания множеств, который подразумевает использование вызова `set`.

3. Как проверить присутствие/отсутствие элемента в множестве?

`<элемент> in <множество>` или `<элемент> not in <множество>`

4. Как выполнить перебор элементов множества?

С помощью цикла `for`

5. Что такое `set comprehension`?

Это метод для создания множеств из других итерируемых объектов

6. Как выполнить добавление элемента во множество?

С помощью метода `add()`

7. Как выполнить удаление одного или всех элементов множества?

Удаление одного элемента производится с помощью метода `remove()`, а удаление при помощи метода `clear()`

8. Как выполнить основные операции над множествами:
объединение, пересечение, разность?

Объединение: `union()`

Пересечение: `intersection()`

Разность: `difference()`

9. Как определить, что некоторое множество является надмножеством или подмножеством другого множества?

С помощью методов `issubset()` и `issuperset()`

10. Каково назначение множеств `frozenset`?

Множество, созданное с помощью этого ключевого слова нельзя изменять.

11. Как осуществляется преобразование множеств в строку, список, словарь?

Для преобразования множества в строку используется конкатенация текстовых значений, которую обеспечивает функция `join`.

Чтобы получить из множества словарь, следует передать функции `dict` набор из нескольких пар значений, в каждом из которых будет находиться ключ.

По аналогии с предыдущими преобразованиями можно получить список неких объектов. На этот раз используется вызов `list`, получающий в качестве аргумента множество `a`.