

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

Кафедра инфокоммуникаций

**Отчет по лабораторной работе №2.7
Работа с множествами в языке Python
по дисциплине «Технологии программирования»**

Выполнил студент группы ИВТ-б-о-20-1

Ищенко Т.С. « » _____ 20__ г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверила Воронкин Р.А. _____

(подпись)

Ставрополь 2021

Цель работы: приобретение навыков по работе с множествами при написании программ с помощью языка программирования Python версии 3.x.

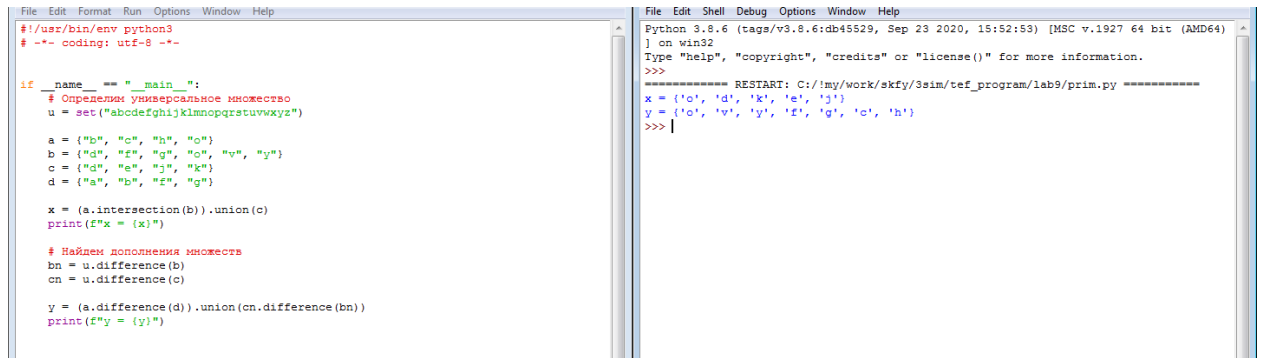
Ссылка на репозиторий: <https://github.com/Timofej8971/lab-9>

Методика и порядок выполнения работы:

1. Изучить теоретический материал работы.
2. Создать общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и язык программирования Python.
3. Выполните клонирование созданного репозитория.
4. Дополните файл .gitignore необходимыми правилами для работы с IDE PyCharm.
5. Организуйте свой репозиторий в соответствие с моделью ветвления git-flow.
6. Создайте проект PyCharm в папке репозитория.
7. Проработайте примеры лабораторной работы. Создайте для них отдельный модуль языка Python. Зафиксируйте изменения в репозитории.
8. Решите задачу: подсчитайте количество гласных в строке, введенной с клавиатуры с использованием множеств.
9. Зафиксируйте сделанные изменения в репозитории.
10. Решите задачу: определите общие символы в двух строках, введенных с клавиатуры.
11. Зафиксируйте сделанные изменения в репозитории.
12. Добавьте отчет по лабораторной работе в формате PDF в папку doc репозитория.
- Зафиксируйте изменения.
13. Выполните слияние ветки для разработки с веткой master/main.
14. Отправьте сделанные изменения на сервер GitHub.

Ход работы:

1. Произвёл выполнение примера



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

if __name__ == "__main__":
    # Определим универсальное множество
    u = set("abcdefghijklmnopqrstuvwxyz")

    a = {"b", "c", "h", "o"}
    b = {"d", "f", "g", "o", "v", "y"}
    c = {"d", "e", "j", "k"}
    d = {"a", "b", "f", "g"}

    x = (a.intersection(b)).union(c)
    print(f"x = {x}")

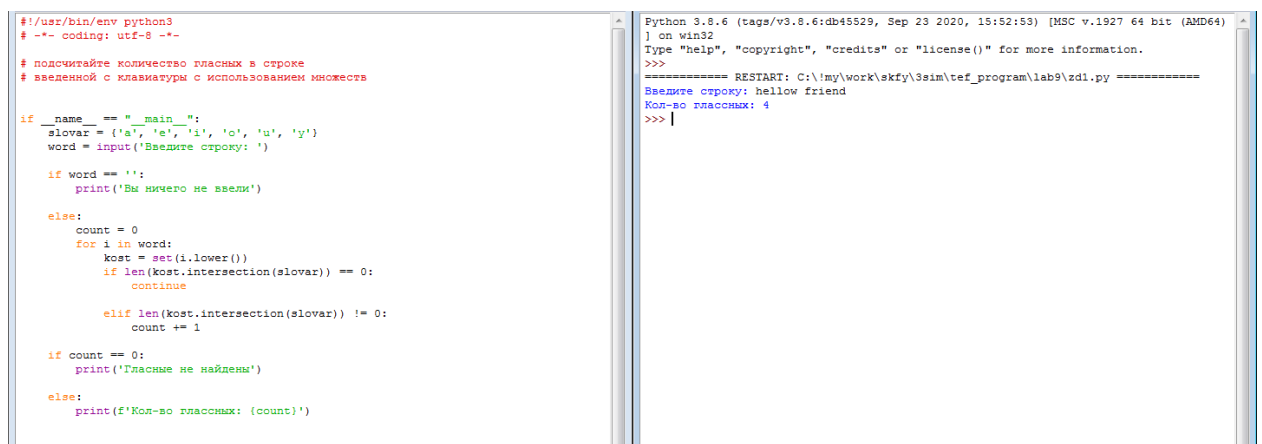
    # Найдём дополнения множеств
    bn = u.difference(b)
    cn = u.difference(c)

    y = (a.difference(d)).union(cn.difference(bn))
    print(f"y = {y}")
```

```
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)]
>>>
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\my\work\skfy\3sim\tef_program\lab9\prim.py =====
x = {'o', 'd', 'k', 'e', 'j'}
y = {'o', 'v', 'y', 'f', 'g', 'c', 'h'}
>>>
```

Рисунок 1 – Результат выполнения примера

2. Выполнил первое задание



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

# подсчитайте количество гласных в строке
# введенной с клавиатуры с использованием множеств

if __name__ == "__main__":
    slovar = {'a', 'e', 'i', 'o', 'u', 'y'}
    word = input('Введите строку: ')

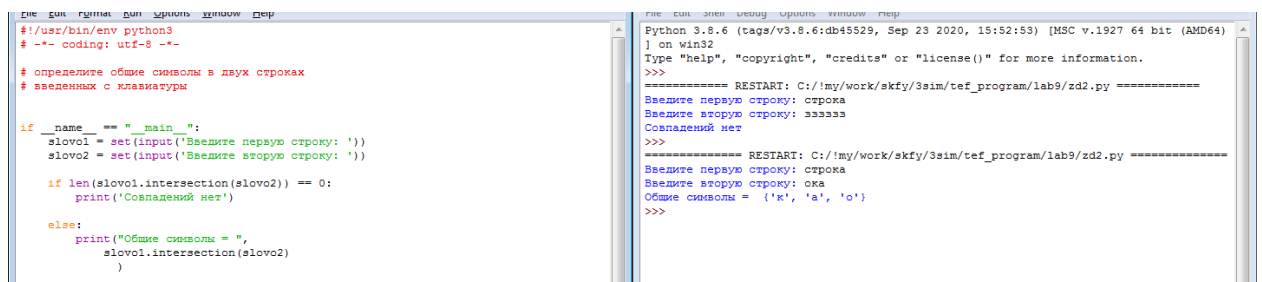
    if word == '':
        print('Вы ничего не ввели')
    else:
        count = 0
        for i in word:
            kost = set(i.lower())
            if len(kost.intersection(slovar)) == 0:
                continue
            elif len(kost.intersection(slovar)) != 0:
                count += 1

        if count == 0:
            print('Гласные не найдены')
        else:
            print(f'Кол-во гласных: {count}')
```

```
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)]
>>>
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\my\work\skfy\3sim\tef_program\lab9\zd1.py =====
Введите строку: hellow friend
Кол-во гласных: 4
>>>
```

Рисунок 2 – Результат выполнения первого задания

3. Выполнил второе задание



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

# определите общие символы в двух строках
# введенных с клавиатуры

if __name__ == "__main__":
    slovo1 = set(input('Введите первую строку: '))
    slovo2 = set(input('Введите вторую строку: '))

    if len(slovo1.intersection(slovo2)) == 0:
        print('Совпадений нет')
    else:
        print("Общие символы = ",
              slovo1.intersection(slovo2))
```

```
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)]
>>>
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\my\work\skfy\3sim\tef_program\lab9\zd2.py =====
Введите первую строку: строка
Введите вторую строку: zzzzzz
Совпадений нет
>>>
===== RESTART: C:\my\work\skfy\3sim\tef_program\lab9\zd2.py =====
Введите первую строку: строка
Введите вторую строку: ока
Общие символы = {'к', 'а', 'о'}
>>>
```

Рисунок 3 – Результат выполнения второго задания

4. Выполнил индивидуальное задание

3. Как проверить присутствие/отсутствие элемента в множестве? – С помощью команды `print(“Искомый элемент” in / not in “Множество”)`

4. Как выполнить перебор элементов множества? – Написав цикл `for “переменная” in “множество”`

5. Что такое `set comprehension`? - генератором, позволяющих заполнять списки, а также другие наборы данных с учетом неких условий.

6. Как выполнить добавление элемента во множество? С помощью функции `add()`

7. Как выполнить удаление одного или всех элементов множества?

— `remove` — удаление элемента с генерацией исключения в случае, если такого элемента нет;

— `discard` — удаление элемента без генерации исключения, если элемент отсутствует;

— `pop` — удаление первого элемента, генерируется исключение при попытке удаления из пустого множества.

8. Как выполняются основные операции над множествами: объединение, пересечение, разность? - Чтобы объединить все элементы двух разных множеств, стоит воспользоваться методом `union` на одном из объектов. Чтобы найти общие элементы для двух разных множеств, следует применить функцию `intersection`, принимающую в качестве аргумента один из наборов данных. Чтобы вычислить разность для двух разных множеств, необходимо воспользоваться методом `difference`.

9. Как определить, что некоторое множество является надмножеством или подмножеством другого множества? - Чтобы выяснить, является ли множество `a` подмножеством `b`, стоит попробовать вывести на экран результат выполнения метода `issubset`, Чтобы узнать, является ли множество `a` надмножеством `b`, необходимо вызвать метод `issuperset` вывести результат его работы на экран.

10. Каково назначение множеств frozenset ? - Множество, содержимое которого не поддается изменению

11. Как осуществляется преобразование множеств в строку, список, словарь? - Для преобразования множества в строку используется конкатенация текстовых значений, которую обеспечивает функция join. Для преобразования в список необходимо использовать функцию list. Для преобразования в словарь необходимо использовать функцию dict.

Вывод: в ходе выполнения лабораторной работы приобрел навыки по работе с множествами при написании программ с помощью языка программирования Python версии 3.x.