

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ ФГАОУ ВО «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ  
УНИВЕРСИТЕТ»

Институт математики и информационных технологии имени  
профессора Н.И. Червякова

Кафедра инфокоммуникаций

**ОТЧЕТ**  
**по лабораторной работе №2**  
Дисциплина: «Языки программирования»

Выполнил: студент 2 курса  
группы ИТС-б-о-20-1  
Новомлинов Алексей Сергеевич  
Проверил доцент  
к.т.н., доцент  
Кафедры инфокоммуникаций  
Воронкин Роман Александрович

Работа защищена с оценкой: \_\_\_\_\_

Ставрополь, 2021

## ЛАБОРАТОРНАЯ РАБОТА №2

### Работа с множествами в языке Python

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

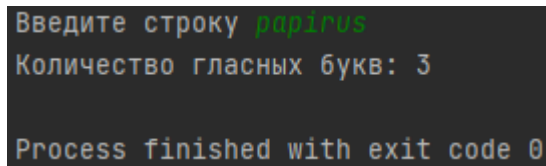
#### Выполнение работы:

Создадим общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT.

[https://github.com/Novomlinov/2C\\_Lab\\_2](https://github.com/Novomlinov/2C_Lab_2)

Напишем код для решения поставленных задач.

**Задание 1.** Подсчитайте количество гласных в строке, введенной с клавиатуры с использованием множеств.

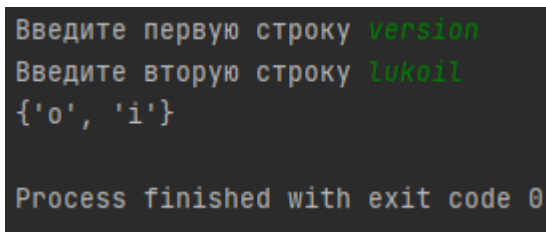


```
Введите строку papius
Количество гласных букв: 3
Process finished with exit code 0
```

Рисунок 1. Результат работы задания 1

Зафиксируем изменения в репозитории.

**Задание 2.** Определите общие символы в двух строках, введенных с клавиатуры.



```
Введите первую строку version
Введите вторую строку lukail
{'o', 'i'}
Process finished with exit code 0
```

Рисунок 2. Результат работы задания 2

Зафиксируем изменения в репозитории.

**Индивидуальное задание (Вариант 12).** Определить результат выполнения операций над множествами. Считать элементы множества строками.

$$A = \{b, k, n, o, q\}; \quad B = \{a, b, k, u\}; \quad C = \{o, p\}; \quad D = \{a, m, n, y, z\};$$

$$X = (A \cup B) \cap D; Y = (\bar{A} \cap D) \cup (C/B).$$

Рисунок 3. Индивидуальное задание

```
x = {'o'}
y = {'a', 'p', 'o', 'z', 'y', 'm'}

Process finished with exit code 0
```

Рисунок 4. Результат работы индивидуального задания

Зафиксируем сделанные изменения в репозитории и отправим на сервер GitHub.

### Контрольные вопросы:

1. Что такое множества в языке Python?

Множеством в языке программирования Python называется неупорядоченная совокупность уникальных значений. В качестве элементов этого набора данных могут выступать любые неизменяемые объекты, такие как числа, символы, строки.

2. Как осуществляется создание множеств в Python?

Сделать это можно, просто присвоив переменной последовательность значений, выделив их фигурными скобками. Существует и другой способ создания множеств, который подразумевает использование вызова `set`. Аргументом этой функции может быть набор неких данных или даже строка с текстом.

3. Как проверить присутствие/отсутствие элемента в множестве?

Чтобы проверить присутствие или отсутствие элемента в множестве, воспользуемся следующей конструкцией:

```
a = {1, 2, 3}
print(3 in a)
```

Если элемент присутствует в множестве, то получим ответ `True`, в противном случае – `False`.

#### 4. Как выполнить перебор элементов множества?

Чтобы выполнить перебор элементов в множестве, воспользуемся следующей конструкцией:

```
for a in {1, 2, 3}
    print(a)
```

#### 5. Что такое set comprehension?

Для создания множества можно в Python воспользоваться генератором, позволяющих заполнять списки, а также другие наборы данных с учетом неких условий.

#### 6. Как выполнить добавление элемента во множество?

Чтобы внести новые значения, потребуется вызывать метод `add`. Аргументом в данном случае будет добавляемый элемент последовательности.

#### 7. Как выполнить удаление одного или всех элементов множества?

Для удаления элементов из множества используются следующие функции в Python:

`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`. Функция позволяет найти элементы, уникальные для второго набора данных, которых в нем нет.

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`. Так как не все элементы набора чисел `a` присутствуют в `b`, функция вернет `False`.

Чтобы узнать, является ли множество `a` надмножеством `b`, необходимо вызвать метод `issuperset` и вывести результат его работы на экран. Поскольку все элементы набора чисел `b` присутствуют в `a`, функция возвращает `True`.

10. Каково назначение множеств `frozenset` ?

Множество, содержимое которого не поддается изменению имеет тип `frozenset`. Значения из этого набора нельзя удалить, как и добавить новые.

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

Для преобразования множества в строку используется конкатенация текстовых значений, которую обеспечивает функция `join`. В этом случае ее аргументом является набор данных в виде нескольких строк. Запятая в кавычках выступает в качестве символа, разделяющего значения. Метод `type` возвращает тип данных объекта в конце приведенного кода.

По аналогии с предыдущими преобразованиями можно получить список неких объектов. На этот раз используется вызов `list`, получающий в качестве аргумента множество `a`. На выходе функции `print` отображаются уникальные значения для изначального набора чисел.

Чтобы получить из множества словарь, следует передать функции `dict` набор из нескольких пар значений, в каждом из которых будет находиться ключ. Функция `print` демонстрирует на экране содержимое полученного объекта, а `type` отображает его тип.

**Вывод:** были приобретены навыки по работе с множествами при написании программ с помощью языка программирования Python версии 3.x.