

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития  
Кафедра инфокоммуникаций

**ОТЧЕТ**  
**ПО ЛАБОРАТОРНОЙ РАБОТЕ №10**  
**дисциплины «Программирование на python»**

Выполнил:  
Кожуховский Виктор Андреевич  
2 курс, группа ИВТ-б-о-22-1,  
09.03.01 «Информатика и  
вычислительная техника»,  
направленность (профиль)  
«Программное обеспечение средств  
вычислительной  
техники и автоматизированных систем  
», очная форма обучения

---

(подпись)

Руководитель практики:  
Воронкин Роман Александрович

---

(подпись)

Отчет защищен с оценкой \_\_\_\_\_ Дата защиты \_\_\_\_\_

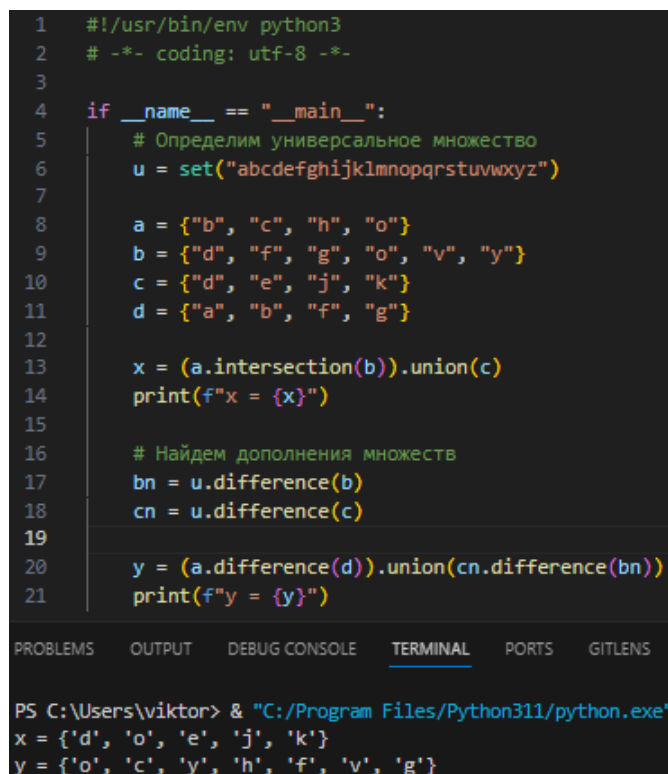
Ставрополь, 2023 г.

**Тема:** Работа с множествами в языке Python

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

### Методика и порядок выполнения работы

1. Изучил теоретический материал работы.
2. Создал общедоступный репозиторий на GitHub, в котором использована лицензия MIT и язык программирования Python.
3. Выполнил клонирование созданного репозитория.
4. Дополнил файл .gitignore необходимыми правилами для работы с IDE PyCharm.
5. Организовал свой репозиторий в соответствии с моделью ветвления git-flow.
6. Создал проект в папке репозитория.
7. Проработал примеры лабораторной работы. Создал для каждого примера отдельный модуль языка Python. Зафиксировал изменения в репозитории.



```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  if __name__ == "__main__":
5      # Определим универсальное множество
6      u = set("abcdefghijklmnopqrstuvwxyz")
7
8      a = {"b", "c", "h", "o"}
9      b = {"d", "f", "g", "o", "v", "y"}
10     c = {"d", "e", "j", "k"}
11     d = {"a", "b", "f", "g"}
12
13     x = (a.intersection(b)).union(c)
14     print(f"x = {x}")
15
16     # Найдём дополнения множеств
17     bn = u.difference(b)
18     cn = u.difference(c)
19
20     y = (a.difference(d)).union(cn.difference(bn))
21     print(f"y = {y}")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

PS C:\Users\viktor> & "C:/Program Files/Python311/python.exe"  
x = {'d', 'o', 'e', 'j', 'k'}  
y = {'o', 'c', 'y', 'h', 'f', 'v', 'g'}

Рисунок 1. Код и выполнение кода из примера 1

8. Решил задачу: подсчитайте количество гласных в строке, введенной с клавиатуры с использованием множеств.

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  if __name__ == "__main__":
5      vowels = set("aeiouAEIOU") # Определение множества гласных
6
7      input_string = input("Введите строку: ") # Ввод строки с клавиатуры
8
9      # Подсчет количества гласных
10     vowel_count = sum(char in vowels for char in input_string)
11
12     print(f"Количество гласных: {vowel_count}")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS SEARCH ERROR

Введите строку: Some big sentence with 8 vowels.  
Количество гласных: 9

Рисунок 2. Код и выполнение кода задания 1

9. Зафиксировал сделанные изменения в репозитории.

10. Решил задачу: определите общие символы в двух строках, введенных с клавиатуры.

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  if __name__ == "__main__":
5      a = set(input("Введите строку a: "))
6      b = set(input("Введите строку b: "))
7
8      x = a.intersection(b)
9      print(f"x = {x}")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\viktor> & "C:/Program Files/Python311/  
Введите строку a: sdaasdfgsefasw  
Введите строку b: asfaewdsawddas  
x = {'s', 'a', 'd', 'f', 'w', 'e'}

Рисунок 3. Код и выполнение кода задания 2

11. Зафиксировал сделанные изменения в репозитории.

Индивидуальное задание: определить результат выполнения операций над множествами. Считать элементы множества строками. Проверить результаты вручную.

$$X = (A/B) \cap (C \cup D); \quad Y = (\bar{A} \cap D) \cup (C/B).$$
$$A = \{b, d, l, p\}; \quad B = \{b, d, e, l, p, x\} \quad C = \{k, l, p, t\}; \quad D = \{d, k, o, p, q, u, v\};$$

```
1  #!/usr/bin/env python3
2  # -*- coding: utf-8 -*-
3
4  if __name__ == "__main__":
5      # Определим универсальное множество
6      u = set("abcdefghijklmnopqrstuvwxyz")
7
8      a = {"b", "d", "l", "p"}
9      b = {"b", "d", "e", "l", "p", "x"}
10     c = {"k", "l", "p", "t"}
11     d = {"d", "k", "o", "p", "q", "u", "v"}
12
13     x = (a.difference(b)).intersection(c.union(d))
14     y = ((u.difference(a)).intersection(d)).union(c.difference(b))
15
16     if x:
17         print(f"x = {x}")
18     else:
19         print("x is empty")
20
21     if y:
22         print(f"y = {y}")
23     else:
24         print(f"y is empty")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS SEARCH ERROR

```
PS C:\Users\viktor> & "C:/Program Files/Python311/python.exe" "c:/Users/viktor
x is empty
y = {'u', 't', 'o', 'v', 'q', 'k'}
```

Рисунок 4. Код и выполнение кода индивидуального задания

12. Добавил отчет по лабораторной работе в формате PDF в папку doc репозитория. Зафиксировал изменения.

13. Выполнил слияние ветки для разработки с веткой main / master.

14. Отправил сделанные изменения на сервер GitHub.

### Вопросы для защиты работы

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. В отличие от массивов и списков, порядок следования значений не учитывается при обработке его содержимого.

2. Как осуществляется создание множеств в Python?

$a = \{1, 2, 0, 1, 3, 2\}$

```
a = set('data')
```

3. Как проверить присутствие/отсутствие элемента в множестве?

Проверка присутствия/отсутствия осуществляется при помощи `in/not in`.

4. Как выполнить перебор элементов множества?

```
for a in {0, 1, 2}:
```

```
    print(a)
```

```
0
```

```
1
```

```
2
```

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.

```
a = {i for i in [1, 2, 0, 1, 3, 2]}
```

```
print(a)
```

```
{0, 1, 2, 3}
```

6. Как выполнить добавление элемента во множество?

Метод `add(аргумент)` в данном случае - добавляемый элемент последовательности.

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.

Метод `clear` очищает массив полностью.

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

Объединение a с b:

```
c = a.union(b)
```

Пересечение:

```
c = a.intersection(b)
```

Разность:

```
c = a.difference(b)
```

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

Подмножества:

```
a = {0, 1, 2, 3, 4}
```

```
b = {3, 2, 1}
```

```
print(a.issubset(b))
```

Надмножества:

```
a = {0, 1, 2, 3, 4}
```

```
b = {3, 2, 1}
```

```
print(a.issuperset(b))
```

10. Каково назначение множеств frozenset ?

Значения из этого набора нельзя удалить, как и добавить новые.

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

В строку:

```
a = {'set', 'str', 'dict', 'list'}
```

```
b = ','.join(a)
```

В словарь:

```
a = {('a', 2), ('b', 4)}
```

```
b = dict(a)
```

В список:

```
a = {1, 2, 0, 1, 3, 2}
```

```
b = list(a)
```