

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»

Вариант 3

Выполнил:
Болуров Ислам Расулович
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 Р. А.

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023

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.

Пример 1. Определить результат выполнения операций над множествами. Считать элементы множества строками.

$$A = \{b, c, h, o\}; \quad B = \{d, f, g, o, v, y\}; \quad C = \{d, e, j, k\}; \quad D = \{a, b, f, g\}; \quad X = (A \cap B) \cup C; \quad Y = (A/D) \cup (\bar{C}/\bar{B}).$$

Примечание: в качестве универсального множества считать все строчные латинские буквы от а до z.

Рисунок 1. Содержание задачи

Листинг программы:

```
#!/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}")

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

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

```
C:\Users\User\PycharmProjects\Python_laba_10\venv\
x = {'e', 'o', 'j', 'k', 'd'}
y = {'o', 'g', 'v', 'y', 'h', 'c', 'f'}

Process finished with exit code 0
```

Рисунок 2. Результат выполнения программы

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

Листинг программы:

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

if __name__ == "__main__":
    glassnie = {'a', 'e', 'i', 'o', 'u', 'y', 'a', 'e', 'ë', 'и', 'o', 'y',
```

```
'ы', 'э', 'ю', 'я'}

s = input("Введите строку: ")

counter = len([i for i in s.lower() if i in glassnie])

print("Количество гласных в строке:", counter)
```

```
C:\Users\User\PycharmProjects\Python_laba_10\venv\
Введите строку: машина
Количество гласных в строке: 3

Process finished with exit code 0
```

Рисунок 3. Тест №1

```
C:\Users\User\PycharmProjects\Python_laba_10\venv\
Введите строку: Alumno
Количество гласных в строке: 3

Process finished with exit code 0
```

Рисунок 4. Тест №2

```
C:\Users\User\PycharmProjects\Python_laba_10\venv\
Введите строку: я олег my name is oleg
Количество гласных в строке: 9

Process finished with exit code 0
```

Рисунок 5. Тест №3

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

Листинг программы:

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

if __name__ == "__main__":
    s_1 = input("Enter the first line: ")
    s_2 = input("Enter the second line: ")

    set1 = set(s_1)
    set2 = set(s_2)

    common_characters = set1.intersection(set2)

    print("Common characters:", common_characters)
```

```
C:\Users\User\PycharmProjects\Python_laba_10\
Enter the first line: я
Enter the second line: семья
Common characters: {'я'}

Process finished with exit code 0
```

Рисунок 6. Тест №1

```
C:\Users\User\PycharmProjects\Python_laba_10\ver
Enter the first line: уравнение
Enter the second line: равнина
Common characters: {'и', 'в', 'р', 'н', 'а'}

Process finished with exit code 0
```

Рисунок 7. Тест №2

```
C:\Users\User\PycharmProjects\Python_laba_10\
Enter the first line: original
Enter the second line: old
Common characters: {'l', 'o'}

Process finished with exit code 0
```

Рисунок 8. Тест №3

Индивидуальное задание. Определить результат выполнения операций над множествами. Считать элементы множества строками.

$$A = \{b, f, g, m, o\}; \quad B = \{b, g, h, l, u\}; \quad C = \{e, f, m\}; \quad D = \{e, g, l, p, q, u, v\};$$

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

Рисунок 9. Содержание индивидуального задания

Листинг программы:

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

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

    a = {"b", "f", "g", "m", "o"}
    b = {"b", "g", "h", "l", "u"}
    c = {"e", "f", "m"}
    d = {"e", "g", "l", "p", "q", "u", "v"}

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

```
print(f"x = {x}")

# Найдём дополнения множества
bn = u.difference(b)

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

```
C:\Users\User\PycharmProjects\Python_laba_10\venv\
x = {'e', 'j', 'k', 'o', 'd'}
y = {'v', 'o', 'c', 'h', 'y', 'g', 'f'}

Process finished with exit code 0
```

Рисунок 10. Результат программы

Вывод: в ходе выполнения данной лабораторной работы были приобретены навыки взаимодействия с множествами при написании программ с помощью языка программирования Python версии 3.x.

Ответы на контрольные вопросы

1. Множество в языке Python - это неупорядоченная коллекция уникальных элементов.
2. Множество можно создать, перечислив элементы в фигурных скобках, разделенных запятыми, например: `my_set = 1, 2, 3`.
3. Для проверки присутствия/отсутствия элемента в множестве используется оператор `in`, например: `if element in my_set:`.
4. Перебор элементов множества можно осуществить с помощью цикла `for`, например: `for element in my_set:`.
5. Set comprehension - это способ создания множества на основе итерации, аналогичный списковому включению, например: `my_set = x for x in range(10)`.
6. Добавление элемента во множество осуществляется с помощью метода `add()`, например: `my_set.add(4)`.
7. Удаление одного или всех элементов множества можно выполнить с помощью методов `remove()` и `clear()` соответственно.
8. Основные операции над множествами в Python:
 - Объединение

- Пересечение

- Разность

9. Для определения надмножества или подмножества используются операторы \leq (является подмножеством) и \geq (является надмножеством).

10. Множества `frozenset` - это неизменяемые множества, которые могут быть использованы в качестве ключей словаря или элементов другого множества.

11. Преобразование множеств в строку, список, словарь осуществляется с помощью функций `str()`, `list()`, `dict()` соответственно.