

Северо-Кавказский федеральный университет  
Институт математики и информационных технологий

**ОТЧЕТ**  
**о выполнении лабораторной работы №4**  
**по дисциплине**  
**«Основы Программной Инженерии»**

Выполнил:

**Маняхин Тимур Александрович**

---

студент 2 курса, ПИЖ-б-о-20-1 группы бакалавриата  
«Программная инженерия»

очной формы обучения

---

Ставрополь, 2021

Северо-Кавказский федеральный университет  
Институт математики и информационных технологий

### Скриншоты программ

```
u = set("abcdefghijklmnopqrstuvwxyz")
A = {"b", "e", "g", "h", "k", "s"}
B = {"c", "g", "p", "q"}
C = {"f", "g", "s", "x", "y", "z"}
D = {"a", "c", "d", "g", "u", "v", "z"}
X = (A.union(B)).intersection(C)
print(f"x = {X}")
an = u.difference(A)
Y = (an.intersection(D)).union(C.difference(B))
print(f"y = {Y}")
```

Рисунок 1.1 – программа iz

```
x = {'g', 's'}
y = {'a', 'c', 'u', 'v', 's', 'd', 'y', 'x', 'f', 'z'}
```

Рисунок 1.2 – результат работы программы iz

## ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

1. 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. Пример 1: `a = {1, 2, 0, 1, 3, 2}` Пример 2: `a = set('data')`
3. С помощью оператора `in`.
4. С помощью цикла `for`.
5. Для создания множества можно в Python воспользоваться генератором, позволяющих заполнять списки, а также другие наборы данных с учетом неких условий.
6. С помощью метода `add()`, `a.add()`
7. 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. • `clear` — удаление всех элементов множества
8. Пример объединения: `c = a.union(b)` Пример пересечения: `c = a.intersection(b)` Пример разности: `c = a.difference(b)`
9. Определение подмножества: `a.issubset(b)` Определение надмножества: `a.issuperset(b)`
10. Множество, содержимое которого не поддается изменению имеет тип `frozenset`. Значения из этого набора нельзя удалить, как и добавить новые. Поскольку содержимое `frozenset` должно всегда оставаться статичным, перечень функций, с которыми такое множество может взаимодействовать, имеет ограничения.
11. Для преобразования множества в строку используется конкатенация текстовых значений, которую обеспечивает функция `join`. Пример: `b = ','.join(a)` Чтобы получить из множества словарь, следует передать функции `dict` набор из нескольких пар значений, в каждом из которых будет находиться ключ. Пример: `a = {'a', 2}, ('b', 4)` `b = dict(a)` Для получения списка: используется вызов `list`, получающий в качестве аргумента множество `a`. Пример: `a = {1, 2, 0, 1, 3, 2}` `b = list(a)`