

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ
ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ» ИНСТИТУТ ЦИФРОВОГО РАЗВИТИЯ

Отчет о лабораторной работе №10
по дисциплине основы программной инженерии

Выполнил: Духно Михаил
Александрович,

2 курс, группа ПИЖ-б-о-20-1,

Проверил: Доцент кафедры
инфокоммуникаций, Воронкин Р.А.

Ставрополь, 2021 г

```
a = {1, 2, 0, 1, 3, 2}
print(a)
#{0, 1, 2, 3}

a = set('data')
print(a)
#{'d', 'a', 't'}
```

Рисунок 10.1 – Создание множеств

```
a = {0, 1, 2, 3}
print(2 in a)
#True

a = {0, 1, 2, 3}
print(2 not in a)
#False

for a in {0, 1, 2}:
    print(a)
#0
#1
#2
```

Рисунок 10.2 – Использование множеств

```
a = {i for i in [1, 2, 0, 1, 3, 2]}
print(a)
#{0, 1, 2, 3}
```

Рисунок 10.3 – Set Comprehensions

```

#Получение размера
a = {0, 1, 2, 3}
print(len(a))
#4

#Добавление элемента
a = {0, 1, 2, 3}
a.add(4)
print(a)
#{0, 1, 2, 3, 4}

#Удаление элемента
a = {0, 1, 2, 3}
a.remove(3)
print(a)
#{0, 1, 2}

#Полная очистка
a = {0, 1, 2, 3}
a.clear()
print(a)
#set()

```

Рисунок 10.4 – Изменение множеств

```

a = {0, 1, 12, 'b', 'ab', 3, 2, 'a'}
print(a)
#{0, 1, 'b', 3, 12, 'ab', 'a'}

a = {0, 1, 12, 3, 2}
print(a)
#{0, 1, 2, 3, 12}

a = {0, 1, 12, 3, 2}
b = list(a)
print(b)
#[0, 1, 2, 3, 12]

```

Рисунок 10.5 – Порядок элементов в множестве

```

#Объединение
a = {0, 1, 2, 3}
b = {4, 3, 2, 1}
c = a.union(b)
print(c)
#{0, 1, 2, 3, 4}

#Добавление
a = {0, 1, 2, 3}
b = {4, 3, 2, 1}
a.update(b)
print(a)
#{0, 1, 2, 3, 4}

```

Рисунок 10.6 – Операции над множествами (1)

```
#Пересечение
a = {0, 1, 2, 3}
b = {4, 3, 2, 1}
c = a.intersection(b)
print(c)
#{1, 2, 3}

#Разность
a = {0, 1, 2, 3}
b = {4, 3, 2, 1}
c = a.difference(b)
print(c)
#{0}
```

Рисунок 10.7 – Операции над множествами (2)

```
#Определение подмножества
a = {0, 1, 2, 3, 4}
b = {3, 2, 1}
print(a.issubset(b))
#False

#Определение надмножества
a = {0, 1, 2, 3, 4}
b = {3, 2, 1}
print(a.issuperset(b))
#True
```

Рисунок 10.8 – Отношения между множествами

```
a = frozenset({'hello', 'world'})
print(a)
#frozenset({'hello', 'world'})
```

Рисунок 10.9 – Тип frozenset

```

#Строка
a = {'set', 'str', 'dict', 'list'}
b = ' '.join(a)
print(b)
print(type(b))
#set dict list str
#<class 'str'>

#Словарь
a = {('a', 2), ('b', 4)}
b = dict(a)
print(b)
print(type(b))
#{'b': 4, 'a': 2}
#<class 'dict'>

#Список
a = {1, 2, 0, 1, 3, 2}
b = list(a)
print(b)
print(type(b))
#[0, 1, 2, 3]
#<class 'list'>

```

Рисунок 10.10 – Преобразование множеств

```

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

if __name__ == '__main__':
    u = set('abcdefghijklmnopqrstuvwxyz')
    a = {'a', 'e', 'f', 'i'}
    b = {'a', 'b', 'k', 'n'}
    c = {'e', 'f', 'n', 'o', 'w', 'x'}
    d = {'a', 'd', 'e', 'o', 'p', 't', 'u'}
    x = c.union(a.intersection(b))
    y = (u.difference(a).intersection(u.difference(b))).difference(c.union(d))
    print(f'x = {x}')
    print(f'y = {y}')

```

Рисунок 10.11 – Код программы индивидуального задания (вариант 8)

```

x = {'w', 'x', 'f', 'a', 'o', 'n', 'e'}
y = {'z', 'q', 's', 'h', 'y', 'r', 'l', 'g', 'm', 'j', 'c', 'v'}

```

Рисунок 10.12 – Результат работы программы

Контрольные вопросы:

1. Что такое множества в языке Python

Множество в языке Питон — это структура данных, эквивалентная множествам в математике. **Множество** может состоять из различных элементов, порядок элементов в **множестве** неопределен.

2. Как осуществляется создание множеств в Python

```
a = {1, 2, 0, 1, 3, 2}
print(a)
#{0, 1, 2, 3}

a = set('data')
print(a)
#{'d', 'a', 't'}
```

3. Как проверить присутствие/отсутствие элемента в множестве

```
a = {0, 1, 2, 3}
print(2 in a)
#True

a = {0, 1, 2, 3}
print(2 not in a)
#False
```

5. Что такое set comprehension

Set comprehensions (генераторы множеств)

Генераторы множеств в целом аналогичны генераторам списков.

```
a = {i for i in [1, 2, 0, 1, 3, 2]}
print(a)
#{0, 1, 2, 3}
```

6. Как выполнить добавление элемента во множество

```
#Добавление элемента
a = {0, 1, 2, 3}
a.add(4)
print(a)
#{0, 1, 2, 3, 4}
```

7. Как выполнить удаление одного или всех элементов множества

```

#Удаление элемента
a = {0, 1, 2, 3}
a.remove(3)
print(a)
#{0, 1, 2}

#Полная очистка
a = {0, 1, 2, 3}
a.clear()
print(a)
#set()

```

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

```

#Объединение
a = {0, 1, 2, 3}
b = {4, 3, 2, 1}
c = a.union(b)
print(c)
#{0, 1, 2, 3, 4}

```

```

#Пересечение
a = {0, 1, 2, 3}
b = {4, 3, 2, 1}
c = a.intersection(b)
print(c)
#{1, 2, 3}

#Разность
a = {0, 1, 2, 3}
b = {4, 3, 2, 1}
c = a.difference(b)
print(c)
#{0}

```

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

```

#Определение подмножества
a = {0, 1, 2, 3, 4}
b = {3, 2, 1}
print(a.issubset(b))
#False

#Определение надмножества
a = {0, 1, 2, 3, 4}
b = {3, 2, 1}
print(a.issuperset(b))
#True

```

10. Какого назначение множеств frozenset

Единственное отличие set от frozenset заключается в том, что set - изменяемый тип данных, а frozenset - нет. Примерно похожая ситуация с [списками](#) и [кортежами](#).

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

```
#Строка
a = {'set', 'str', 'dict', 'list'}
b = ' '.join(a)
print(b)
print(type(b))
#set dict list str
#<class 'str'>

#Словарь
a = {('a', 2), ('b', 4)}
b = dict(a)
print(b)
print(type(b))
#{'b':4, 'a':2}
#<class 'dict'>

#Список
a = {1, 2, 0, 1, 3, 2}
b = list(a)
print(b)
print(type(b))
#[0, 1, 2, 3]
#<class 'list'>
```