

Робота з масивами (2 част.)

Словники

Словник – це невідсортована змінна структура даних. У елементів словника немає індексів, але вони мають ключі. Словники записуються у фігурних дужках за типом: “ключ : елемент”.

```
slov = {'2':'dog', '1':'cat', '4':'hamster', '7':'crocodile'}
```

Функція конвертації – dict().

Пошук по словнику відбувається наступним чином:

```
>>> slov['4']  
'hamster'
```

Так як словники є змінними, то в них можна додати, видалити або змінити елемент.

Щоб додати елемент:

```
>>> slov['5']='elephant'  
>>> slov['5']  
'elephant'
```

Змінити його:

```
>>> slov['7']='zubastyk'  
>>> slov['7']  
'zubastyk'
```

Або видалити:

```
>>> del slov['5']  
>>> slov  
{'2': 'dog', '1': 'cat', '4': 'hamster', '7': 'zubastyk'}
```

У одному словнику не може бути двох елементів з однаковими ключами, але можуть бути однакові елементи, які належать різним ключам.

Множини

Множини – це масиви неповторюваних невідсортованих елементів. При конвертації, наприклад, списку, усі дублікати елементів видаляються автоматично. Множини можуть бути змінюваними (set) або незмінюваними (frozenset). Їхня різниця аналогічна до різниці між списком та кортежем.

Множини записуються у фігурних дужках, але створення пустої множини записом дужок неможливе, т.я. в такому випадку створиться порожній словник. Множина записується за допомогою конвертації списку, кортежу або строку або вказанням порожньої функції:

```
>>> array = set()
>>> array
set()
```

```
>>> array = set([1,3,5,4,4,7])
>>> array
{1, 3, 4, 5, 7}
```

При вводиті рядка у функцію set() створиться множина символів цього рядка:

```
>>> array = set('hello, world!')
>>> array
{'w', 'r', 'd', 'l', 'h', 'e', ',', ' ', '!', 'o'}
```

Для додання до множини елементу використовується метод set.add(), в дужках якого вказують елемент, що додається:

```
>>> array = set()
>>> array
set()
>>> array.add(4)
>>> array
{4}
```

Для видалення елементу використовують один з наступних методів:

- set.remove() – прибирає елемент, зазначений в дужках. Помилка (KeyError), якщо елемента немає в множині.
- set.discard() – прибирає елемент, зазначений в дужках. Якщо елемента немає, не повертає помилку.
- set.pop() – прибирає випадковий елемент з множини;
- set.clear() – очищує множину.

Також над множинами можливо виконувати логічні операції:

- & – логічне І;
- | – логічне АБО;
- - – логічне віднімання;
- ^ – імплікація.

Для змінення вже існуючої множини використовують ті самі дії, але зі знаком =:

```
>>> array = {4,5,6}
>>> arr = {5,6,7,8}
>>> array &= arr
>>> array
{5, 6}
```

```
>>> array |= arr
>>> array
{5, 6, 7, 8}
```

Контрольні питання:

1. У чому особливість словника?
2. Як взяти елемент з словника?
3. У чому особливість множини?
4. Які дії можна виконувати з множинами?