

федеральное государственное автономное образовательное учреждение  
высшего образования  
Санкт-Петербургский национальный исследовательский университет  
информационных технологий, механики и оптики

Лабораторная работа №7  
по дисциплине: «Теория Алгоритмов»  
Тема: «Хэш-таблицы»

Выполнила ученица 223 группы  
Никончук А.П.

Проверил  
Сорокин Д.С. \_\_\_\_\_

Санкт-Петербург,  
2015 г.

## Открытое хэширование

1. Написать функцию `set_item()`, которая принимает в качестве аргументов ключ `key` и значение `value`, которое по этому ключу необходимо разместить в словаре.

```
def set_item_separate(table, key, value):
    i = hash_code(key, table)
    if table[i] == None:
        list1 = []
        list1.append(key)
        list1.append(value)
        table[i] = list1
    else:
        list2 = [table[i]]
        list2.append([key, value])
        table[i] = list2
    return table
```

```
set_item_separate(table, 'Arnold', 7)
set_item_separate(table, 'Helga', 7)
set_item_separate(table, 'Gerald', 8)
set_item_separate(table, 'Phoebe', 9)
set_item_separate(table, 'Sid', 8)
set_item_separate(table, 'Stinky', 2)
print(table)
```

Администратор@TATYANNA ~/cs101myworks/2сем/7 (master)

\$ python hash.py

```
[[['Phoebe', 9], ['Sid', 8], None, ['Gerald', 8], None, [['Stinky', 2], ['Helga', 7]], ['Arnold', 7]]
```

2. Написать функцию `get_item()`, которая возвращает значение по ключу `key`. Если такого ключа в словаре не существует, то необходимо создать исключение `KeyError()`:

```
def get_item_separate(table, key):
    i = hash_code(key, table)
    for j in range(len(table[i])):
        if key not in table[i][j]:
            raise KeyError((key) + ' is not in table')
        if key in table[i][j]:
            value = table[i][j][1]
            return value
```

```
set_item_separate(table, 'Arnold', 7)
set_item_separate(table, 'Helga', 7)
set_item_separate(table, 'Gerald', 8)
set_item_separate(table, 'Phoebe', 9)
set_item_separate(table, 'Sid', 8)
set_item_separate(table, 'Stinky', 2)
print(table)
print(get_item_separate(table, 'Helga'))
```

Администратор@TATYANNA ~/cs101myworks/2сем/7 (master)

\$ python hash.py

```
[[['Phoebe', 9], ['Sid', 8], None, ['Gerald', 8], None, [['Helga', 7], ['Stinky', 2]], ['Arnold', 7]]
7
```

```
set_item_separate(table, 'Arnold', 7)
set_item_separate(table, 'Helga', 7)
set_item_separate(table, 'Gerald', 8)
set_item_separate(table, 'Phoebe', 9)
set_item_separate(table, 'Sid', 8)
set_item_separate(table, 'Stinky', 2)
print(table)
print(get_item_separate(table, 'Eugene'))
```

Администратор@TATYANNA ~/cs101myworks/2сем/7 (master)

\$ python hash.py

```
[[['Phoebe', 9], ['Sid', 8], None, ['Gerald', 8], None, [['Helga', 7], ['Stinky', 2]], ['Arnold', 7]]
```

Traceback (most recent call last):

File "hash.py", line 91, in <module>

print(get\_item\_separate(table, 'Eugene'))

File "hash.py", line 34, in get\_item\_separate

raise KeyError((key) + ' is not in table')

KeyError: 'Eugene is not in table'

3. Написать функцию `delete_item()`, которая удаляет элемент с ключом `key` из словаря. Аналогично работе функции `get_item()` должно возникать исключение `KeyError()` при попытке удалить ключ, которого нет в словаре

```
def delete_item_separate(table, key):
    i = hash_code(key, table)
    if table[i] == None:
        raise KeyError((key) + ' is not in table')
        return table
    else:
        if type(table[i][0]) == list:
            old = get_item_separate(table, key)
            table[i].append(old)
        else:
            raise KeyError((key) + ' is not in table')
        return table
```

```
set_item_separate(table, 'Arnold', 7)
set_item_separate(table, 'Helga', 7)
set_item_separate(table, 'Gerald', 8)
set_item_separate(table, 'Phoebe', 9)
set_item_separate(table, 'Sid', 8)
set_item_separate(table, 'Stinky', 2)
print(table)
delete_item_separate(table, 'Arnold')
print(table)
```

```
Администратор@TATYANNA ~/cs101myworks/2сем/7 (master)
$ python hash.py
[['Phoebe', 9], ['Sid', 8], None, ['Gerald', 8], None, [['Helga', 7], ['Stinky', 2]], ['Arnold', 7]]
[['Phoebe', 9], ['Sid', 8], None, ['Gerald', 8], None, [['Helga', 7], ['Stinky', 2]], None]
```

```
set_item_separate(table, 'Arnold', 7)
set_item_separate(table, 'Helga', 7)
set_item_separate(table, 'Gerald', 8)
set_item_separate(table, 'Phoebe', 9)
set_item_separate(table, 'Sid', 8)
set_item_separate(table, 'Stinky', 2)
print(table)
delete_item_separate(table, 'Eugene')
print(table)
```

```
Администратор@TATYANNA ~/cs101myworks/2сем/7 (master)
$ python hash.py
[['Phoebe', 9], ['Sid', 8], None, ['Gerald', 8], None, [['Helga', 7], ['Stinky', 2]], ['Arnold', 7]]
Traceback (most recent call last):
  File "hash.py", line 86, in <module>
    delete_item_separate(table, 'Eugene')
  File "hash.py", line 26, in delete_item_separate
    raise KeyError((key) + ' is not in table')
KeyError: 'Eugene is not in table'
```

4. Написать генераторы `iterate_by_keys()` и `iterate_by_pairs()`, которые генерируют ключи и пары ключ-значение, соответственно.

```
def iterate_by_keys_separate():
    for i in table:
        if i != None:
            if type(i[0])==List:
                for j in i:
                    yield j[0]
            else:
                yield i[0]

set_item_separate(table,'Arnold',7)
set_item_separate(table,'Helga',7)
set_item_separate(table,'Gerald',8)
set_item_separate(table,'Phoebe',9)
set_item_separate(table,'Sid',8)
set_item_separate(table,'Stinky',2)
print(table)

gen=iterate_by_keys_separate()

print(next(gen))
print(next(gen))
print(next(gen))
print(next(gen))
print(next(gen))
print(next(gen))
```

```
Администратор@TATYANNA ~/cs101myworks/2cem/7 (master)
$ python hash.py
[['Phoebe', 9], ['Sid', 8], None, ['Gerald', 8], None, [['Helga', 7], ['Stinky',
2]], ['Arnold', 7]]
Phoebe
Sid
Gerald
Helga
Stinky
Arnold
```

```
def iterate_by_pairs_separate():
    for i in table:
        if i != None:
            if type(i[0])==List:
                for j in i:
                    yield str(j[0])+' '+str(j[1])
            else:
                yield str(i[0])+' '+str(i[1])

set_item_separate(table,'Arnold',7)
set_item_separate(table,'Helga',7)
set_item_separate(table,'Gerald',8)
set_item_separate(table,'Phoebe',9)
set_item_separate(table,'Sid',8)
set_item_separate(table,'Stinky',2)
print(table)

gen=iterate_by_pairs_separate()

print(next(gen))
print(next(gen))
print(next(gen))
print(next(gen))
print(next(gen))
print(next(gen))
```

```
Администратор@TATYANNA ~/cs101myworks/2cem/7 (master)
$ python hash.py
[['Phoebe', 9], ['Sid', 8], None, ['Gerald', 8], None, [['Helga', 7], ['Stinky',
2]], ['Arnold', 7]]
Phoebe 9
Sid 8
Gerald 8
Helga 7
Stinky 2
Arnold 7
```