



Что под капотом у Python

а также сложности,
typing и venv



Темы лекции

- сложность
- что под капотом у Python
- typing
- виртуальное окружение

Что такое сложность?

Сложность - это количество ресурсов, которое программа должна потратить для решения какой-то задачи.

Нотация O (о большое) - это ограничение асимптотической сложности сверху, с точностью до константного множителя.

Пояснение про константный множитель:

$$O(n \cdot C)^2 = O(n^2)$$

Примеры

$O(1)$

$O(N)$

$O(N^2)$

Почему важно помнить про сложность?

- Чем меньше сложность, тем эффективнее программа.
- Компромисс между сложностью по памяти и времени.

Реализации Python

- CPython
- Jython
- PyPy
- IronPython

PyObject

```
typedef struct _object {  
    _PyObject_HEAD_EXTRA  
    Py_ssize_t ob_refcnt; // количество ссылок на объект  
    PyTypeObject *ob_type; // указатель на тип объекта  
} PyObject;
```

List

- Список состоит из некоторого количества элементов.
- Элементы могут иметь разный тип.
- Элементы могут повторяться.
- Список в питоне представляет собой ни что иное как динамический массив.

List в CPython

```
typedef struct {
```

```
    PyObject_VAR_HEAD
```

```
    PyObject **ob_item; // массив указателей на элементы списка
```

```
    Py_ssize_t allocated; // размер выделенной памяти под список
```

```
} PyListObject;
```

Таблица сложностей для List

append	индексация	pop	insert	remove	поиск по значению
$O(1)$	$O(1)$	$O(1)$	$O(N)$	$O(N)$	$O(N)$

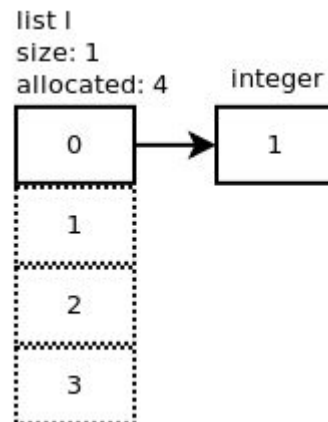
Почему append за $O(1)$?

Память для хранения списка изменяется не при каждом обновлении, а только если выполняется определенное условие для нового количества элементов в списке $(\text{newsize} \gg 3) + (\text{newsize} < 9 ? 3 : 6) = 3$

т.е. для количества элементов

0, 4, 8, 16, 25, 35, 46, 58, 72, 88 ...

[Где почитать подробнее](#)



Dict

- Словарь состоит из элементов ключ-значение.
- Ключи уникальны, значения любые.
- Ключи должны быть хэшируемыми объектами.

Хэш-таблица

- Хэш-функция
- Коллизии
- Метод `__hash__()`

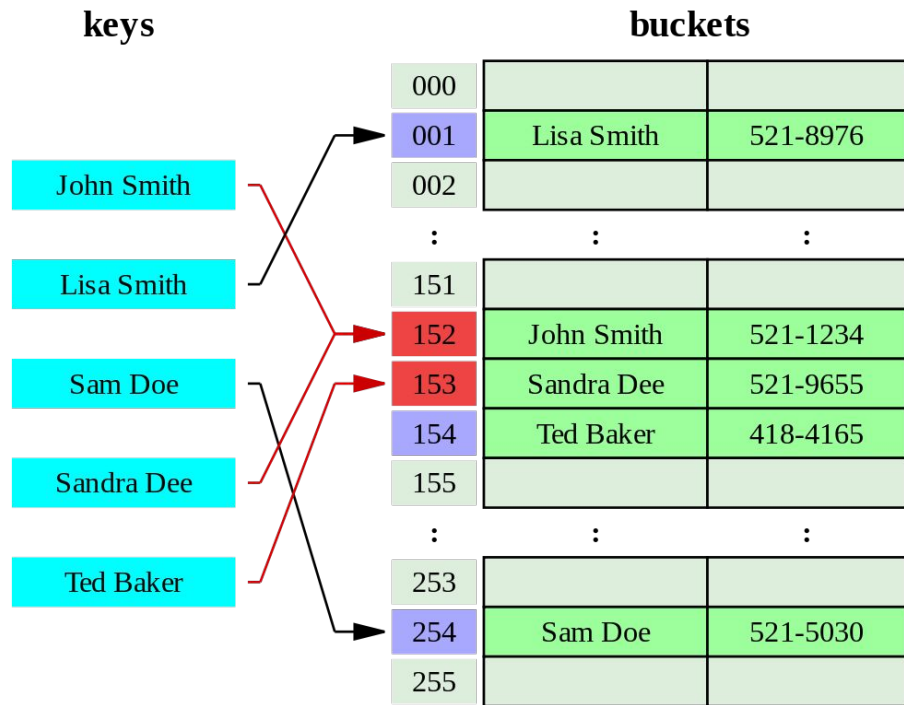


Таблица сложностей для Dict

Запись	Поиск элемента по ключу	Удаление
$O(1)$	$O(1)$	$O(1)$

Set

- Реализовано на основе хэш-таблицы.
- Содержит только уникальные элементы.

Таблица сложностей для Set

Проверка вхождения элемента	Добавление	Удаление
$O(1)$	$O(1)$	$O(1)$

Typing

Что и зачем?

```
def (a, b, c, d = []):  
    pass
```

А как понять, какой тип переменной мы ожидаем?

Примеры со встроенным typing

```
def (a: str, b: int, c: tuple, d: list = []) -> None:
```

```
    pass
```

Примеры с typing

from typing import Optional, Any, Dict, List # и другие

[Документация и описание типов](#)

Виртуальное окружение

Что это и зачем оно надо?

- исключаем конфликты зависимостей
- ограждаем основную систему от мусорных пакетов

Создание виртуального окружения

```
python3 -m venv venv_name
```

Активация/деактивация venv

source venv_name/bin/activate # активировали виртуальное окружение

deactivate # деактивировали

Установка пакетов

pip install some_package_name

pip install -r requirements

Проверка установленных пакетов

pip list # выведет список установленных пакетов

albumentations 0.4.6

aniso8601 9.0.1

apturl 0.5.2

pip freeze # выведет список установленных пакетов в формате requirements

albumentations==0.4.6

aniso8601==9.0.1

apturl==0.5.2

Контакты

Голубева Валентина: valentina.golubeva@moevm.info

Шевская Наталья Владимировна: natalya.shevskaya@moevm.info

Иванов Дмитрий Владимирович: dmitry.ivanov@moevm.info

Префикс в теме письма [CS_23XX]

Преподаватели лабораторных

2300 -- Валентина Петровна Голубева valentina.golubeva@moevm.info

2303 -- Роман Павлович Шестопалов roman.shestopalov@moevm.info

2304 -- Роман Павлович Шестопалов roman.shestopalov@moevm.info

2381 -- Максим Олегович Доброхвалов maxim.dobrokhvalov@moevm.info

2382 -- Максим Олегович Доброхвалов maxim.dobrokhvalov@moevm.info

2383 -- Сергей Валерьевич Сорокумов sergey.sorokumov@moevm.info

2384 -- Сергей Валерьевич Сорокумов sergey.sorokumov@moevm.info

В копию можно ставить: Дмитрий Владимирович Иванов dmitry.ivanov@moevm.info