

Python



План занятия

- Работа с командной строкой
- "Сложные" типы данных
- Функции

Работаем с КОНСОЛЬЮ

Консоль на Windows и Unix разная

[https://access.redhat.com/documentation/en-US/
Red_Hat_Enterprise_Linux/4/html/Step_by_Step_Guide/ap-doslinux.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/4/html/Step_by_Step_Guide/ap-doslinux.html)

Что такое PATH?

- В 'PATH' хранится список путей, где система ищет исполняемые файлы
- 'PATH' можно редактировать

Windows: <http://www.computerhope.com/issues/ch000549.htm>

Unix: <http://www.cyberciti.biz/faq/unix-linux-adding-path/>

Какие бывают частые проблемы при работе с консолью?

- В имени пути есть пробелы или русские буквы
- Программы, которую мы вызываем, нет в PATH
- linux'о-проблемы

Практическая задача

- Посмотрим, где мы сейчас находимся, посмотрим список файлов и папок в текущей папке
- Создадим папку "tasks", посмотрим обновленный список файлов и папок
- Перейдем в папку "tasks"
- Создадим файл "task1.py"
- Отредактируем его, добавим текст ``print('I am python!')``, сохраним изменения
- Выполним файл "task1.py"
- Удалим файл "task1.py"
- Вернемся в начальную папку

"Сложные" типы данных

Неизменяемые типы (immutable):

- Кортежи: tuple

Изменяемые типы (mutable):

- Списки: list
- Словари: dict

«Присваивание никогда не копирует
данные!».

—Знающие люди

Практика

`complex_types/*.py`

Какова сложность операций?

<https://wiki.python.org/moin/TimeComplexity>

Что такое O()?

https://ru.wikipedia.org/wiki/%C2%ABO%C2%BB_%D0%B1%D0%BE%D0%BB%D1%8C%D1%88%D0%BE%D0%B5_%D0%B8_%C2%ABo%C2%BB_%D0%BC%D0%B0%D0%BB%D0%BE%D0%B5

<https://habrahabr.ru/post/188010/>

Классические алгоритмы

Мы все когда-то хотели работать в Google

Сортировка пузырьком

https://en.wikipedia.org/wiki/Bubble_sort

- Дан исходный массив
- Предполагаем, что он не отсортирован
- До тех пор, пока массив не отсортирован, начиная с первого члена, сравниваем его со следующим
- Если все члены идут по возрастанию, заканчиваем работу программы
- Иначе, меняем местами два члена

Бинарный поиск

- Дан исходный отсортированный массив, производим в нем поиск
- Сначала смотрим на член по центру
- Если нужно значение меньше - смещаемся влево
- Если нужно значение больше - смещаемся вправо

Практика

- Дан массив [1, 3, 4, 10, 9, 11, -9, 0, -100]
- Написать сортировку пузырьком массива по возрастанию
- Написать бинарный поиск для данного массива

Функция в Python

```
def my_function(input_var1, input_var2):  
    print(input_var1, input_var2)  
    return input_var1 + input_var2
```

Где:

`def` – ключевое слово для объявления функции

`my_function` – имя функции (бывают функции без имен)

`input_var1, input_var2` – входные параметры

`return` – ключевое слово, чтобы вернуть значение туда, где функция была вызвана

«Зачем нам нужны функции?»

– Не знающие люди

Практика

`course2/functions.py`