

# Пути

## Описание системы

---

### Класс

Алгоритм - класс, содержащий некоторую функцию обрабатывающую объекты

- Вход - один объект
- Выход - список объектов

Спецификация - задекларированные (!= фактические) возможности алгоритма. Представляет собой словарь

- Ключ - класс принимаемого объекта
- Значение - список классов объектов получаемых в результате

Путь - последовательность из классов, которые могут быть получены в результате работы алгоритмов

Строковое представление пути - имена классов входящих в путь через слеш

Компонент - совокупность нескольких алгоритмов, реализующий циклическую обработку объектов

### Реализация класса Component

```
from itertools import chain

class Component:
    def __init__(self, *algorithm_list):
        self.algorithm_list = algorithm_list

    def __call__(self, source_object):
        result = []
        queue = [source_object]

        while queue:
            result.extend(queue)

            queue = list(chain.from_iterable(
                algorithm(item)
                for item in queue
                for algorithm in self.algorithm_list
            ))

        return result
```

## Пример классов и алгоритмов

```
from itertools import chain

class Apple:
    pass

class Orange:
    def __init__(self, number):
        self.number = number

class Lemon:
    pass

class FirstAlgorithm:
    SPECIFICATION = {
        Orange: [Apple],
        Lemon: [Orange, Apple]
    }

    def __call__(self, source_object):
        if isinstance(source_object, Orange):
            return [
                Apple()
                for _ in range(source_object.number)
            ]

        if isinstance(source_object, Lemon):
            return [Orange(3), Apple()]

        return []

class EmptyAlgorithm:
    SPECIFICATION = {}

    def __call__(self, source_object):
        return []
```

# Задача

---

Реализовать новый метод для класса Component

- Вход - класс объекта
- Выход - возможности компонента, если он будет вызван с объектом переданного класса

Формат описания возможностей:

- Потенциальные пути - список строковых представлений потенциально возникающих (изначально переданных / созданных при работе) путей
- Применяемые возможности алгоритмов - словарь
  - Ключ - имя класса алгоритма
  - Значение - словарь
    - Ключ - строковое представление принимаемого пути
    - Значение - список строковых представлений выдаваемых путей

Пример результата:

```
import json

component = Component(FirstAlgorithm(), EmptyAlgorithm())

print(json.dumps(
    component.my_method(Lemon),
    indent=4
))

output:
{
  'Potential': [
    '/Lemon',
    '/Lemon/Orange',
    '/Lemon/Apple'
    '/Lemon/Orange/Apple'
  ],
  'Algorithm': {
    'FirstAlgorithm': {
      '/Lemon': [
        '/Lemon/Orange',
        '/Lemon/Apple'
      ],
      '/Lemon/Orange': [
        '/Lemon/Orange/Apple'
      ]
    },
    'EmptyAlgorithm': {}
  }
}
```