

Лабораторная 8. Вариант 1.

Задача. Фильтр анаграмм

Описание

Анаграммы — это слова, которые состоят из одинаковых букв.

Например:

спаниель — апельсин

карат — карта — катар

топор — ропот — отпор

Формулировка задачи

Реализуйте функцию `filter_anagrams()`, которая находит все анаграммы.

Функция принимает исходный объект и последовательность (iterable) объектов для проверки, а возвращает последовательность анаграмм.

Строго говоря, ваша функция должна уметь находить анаграммы любых последовательностей, в том числе списков и кортежей. То есть решение должно быть максимально общим.

Входные данные

Первый аргумент: элемент (строка, список, кортеж)

Второй аргумент: последовательность элементов в виде списка

Выходные данные

Последовательность анаграмм

Пример 1

Входные данные

```
list(filter_anagrams('abba', ['aabb', 'abcd', 'bbaa', 'dada']))
```

Выходные данные

```
['aabb', 'bbaa']
```

Пример 2

Входные данные

```
list(filter_anagrams('racer', ['crazer', 'carer', 'racar', 'caers', 'racer']))
```

Выходные данные

```
['carer', 'racer']
```

Пример 3

Входные данные

```
list(filter_anagrams('laser', ['lazing', 'lazy', 'lacer']))
```

Выходные данные

```
[]
```

Пример 4

Входные данные

```
list(filter_anagrams([1, 2], [[2, 1], [2, 2], [1, 2]]))
```

Выходные данные

```
[[2, 1], [1, 2]]
```

Примечание

Задание оценивается в 3 балла. Если задача не решается с использованием одной или нескольких функций (lambda, map, filter, reduce), то отнимается 1 балл.

Дополнительные тесты

Файл `main.py` проверяется с помощью линтера [super_linter](#). При проверке игнорируются ошибки D, S, I.