

Лабораторная 7. Вариант 4.

Задача. Передвинуть в конец

Описание

В данной лабораторной работе вам необходимо реализовать две функции. Одна обычным способом, другая рекурсивным.

Также нужно создать декоратор, который будет отслеживать время работы каждой функции и эмпирически показать, какой из способов является оптимальным.

В отдельном файле test.txt прописать минимум 10 всевозможных случаев(рассмотреть также частные случаи), включающих проверку как для больших, так и маленьких по длине или значению входных данных.

Формат записи, следующий:

```
Случай 1
#Обычная функция
Название функции:
Аргументы:
Время выполнения: __ сек.
Результат:
```

```
#Рекурсивная функция
Название функции:
Аргументы:
Время выполнения: __ сек.
Результат:
```

```
...
Случай N
...
```

Формулировка задачи

Напишите функцию, которая перемещает элементы одного «типа» в конец списка.

Функция получает на вход список `my_list` и шаблон элементов `element`, которые нужно переместить в конец.

Функция должна вернуть измененный список. Если элемент отсутствует в списке или возникновение частных случаев, выводить `False`.

Входные данные

Список, элемент списка

Выходные данные

Список

Пример 1

Входные данные

```
move_to_end([1, 3, 2, 4, 4, 1], 1)
```

Выходные данные

```
[3, 2, 4, 4, 1, 1]
```

Пример 2

Входные данные

```
move_to_end([7, 8, 9, 1, 2, 3, 4], 9)
```

Выходные данные

```
[7, 8, 1, 2, 3, 4, 9]
```

Пример 3

Входные данные

```
move_to_end(["a", "a", "a", "b"], "a")
```

Выходные данные

```
["b", "a", "a", "a"]
```

Дополнительные тесты

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