

Теория алгоритмов, структуры данных и сопряженные вопросы

Содержание

1	Задачи	1
1.1	Объединение последовательностей целых чисел в диапазоны	1
1.2	Удаление элемента из массива за линейное время	2
	Список литературы	3
	Список листингов	3

1. Задачи

1.1. Объединение последовательностей целых чисел в диапазоны

Пусть есть массив неотрицательных целых чисел [3, 0, 1, 2, 5, 8, 9, 15]. Требуется построить диапазоны вида "0-3,5,8-9,15".

```
import typing as t

def seq_to_ranges(seq: t.Sequence[int]) -> str:
    """Объединяет элементы последовательности в диапазоны"""

    # принцип раннего отказа
    # если последовательность пустая, возбуждаем исключение
    assert seq, "Error! Empty sequence ..."

    if len(seq) == 1:
        return str(seq[0])

    seq: t.List[int] = sorted(set(seq))

    groups: t.List[t.List[int]] = []
    group: t.List[int] = [seq[0]]

    for value in seq:
        if value == group[-1]:
            continue

        if (value - group[-1]) == 1:
            group.append(value)
            # если разность != 1, значит группа закончилась
        else:
            groups.append(group)
            group = [value]
    groups.append(group) # NB!

    _result: t.List[str] = []
    for group in groups:
        if len(group) > 1:
            _result.append(f"{group[0]}-{group[-1]}")
```

```

        else:
            _result.append(str(group[0]))

    return ",".join(_result)

# Использование
seq = (7,)
seq_to_ranges(seq) # '7'

seq = (1, 2, 3)
seq_to_ranges(seq) # '1-3'

seq = (1, 5, 2, 8, 15, 16, 20, 17)
seq_to_ranges(seq) # '1-2,5,8,15-17,20'

seq = [1, 2, 3, 5, 6, 7, 10, 11, 12, 15, 20]
seq_to_ranges(seq) # '1-3,5-7,10-12,15,20'

```

1.2. Удаление элемента из массива за линейное время

Дан массив целых чисел. Требуется удалить заданный элемент <https://programforyou.ru/poleznoe/how-to-remove-values-from-array-effectively>

Решение на Python

```

import typing as t

def remove_elem(seq: t.Sequence[int], remove_value: int) -> t.List[int]:
    """Удаляет элемент по значению"""
    # в Python параметры в функцию можно передать только по соиспользованию;
    # на практике это означает, что формальные параметры функции всегда получают
    # копии ссылок на фактические аргументы, поэтому если объект изменяемый, то
    # его можно изменить из-под функции;
    # чтобы не изменять список в глобальной области видимости,
    # приходится создавать копию списка с помощью list()
    seq: t.List[int] = list(seq)

    if (value not in seq):
        raise ValueError(
            f"Ошибка! Указанное значение ({value}) "
            "не встречается среди элементов списка"
        )

    j = 0
    for value in seq:
        # если текущий элемент не совпадает с удаляемым, то элемент
        # помещается на свое же место и индекс инкрементируется;
        # в противном случае индекс элемента не инкрементируется и потому
        # следующий элемент встает на место удаленного
        if (value != remove_value):
            seq[j] = value
            j += 1

    # итоговый список будет очевидно на один элемент короче,
    # поэтому нужно забрать все элементы кроме последнего
    return seq[:-1]

# Использование
seq = [10, 8, -5, 6, 0, 3]

```

```
remove_elem(seq, remove_value=6) # [10, 8, -5, 0, 3]

seq = (5, 3, 0, 6)
remove_elem(seq, remove_value=3) # [5, 0, 6]
```

Список литературы

1. *Кольцов Д.М.* Си на примерах. Практика, практика и только практика. – СПб.: Наука и Техника, 2019. – 288 с.

Листинги