

Несколько человек стоят в ряд, разделенные на две команды. Первый человек переходит в команду 1, второй — в команду 2, третий — в команду 1 и так далее.

Задача

Учитывая массив положительных целых чисел (веса людей), верните новый кортеж из двух целых чисел, где первое — это общий вес команды 1, а второе — общий вес команды 2. Считается что массив никогда не будет пустым.

Даны две строки, *a* и *b*, верните строку вида: [короткая строка] + [развернутая длинная]+[короткая строка].

Другими словами, самая короткая строка должна быть указана как префикс и суффикс обратной строки самой длинной.

Если *a* и *b* имеют одинаковую длину, рассматривайте *a* как более длинный, создавая *b*+развернутая *a*+*b*

Дан массив цифр, верните массив, в котором каждая цифра увеличивается на ее позицию в массиве: первая цифра будет увеличена на 1, вторая цифра на 2 и т. д. Обязательно начните отсчитывать свои позиции с 1 (а не 0).

Ваш результат может содержать только однозначные числа, поэтому, если сложение цифры с ее позицией дает вам многозначное число, должна быть возвращена только последняя цифра числа.

Примечания:

- вернуть пустой массив, если переданный массив пуст
- массивы будут содержать только числа, поэтому не беспокойтесь об их проверке

Возьмите следующий IPv4-адрес: 128.32.10.1. Этот адрес состоит из 4 октетов, каждый из которых представляет собой один байт (или 8 бит).

- 1-й октет 128 имеет двоичное представление: 10000000.
- 2-й октет 32 имеет двоичное представление: 00100000.
- 3-й октет 10 имеет двоичное представление: 00001010.
- 4-й октет 1 имеет двоичное представление: 00000001.

Итак 128.32.10.1 == 10000000.00100000.00001010.00000001

Поскольку приведенный выше IP-адрес имеет 32 бита, мы можем представить его как 32-битное число: 2149583361.

Напишите функцию `ip_to_int32(ip)`, которая принимает адрес IPv4 и возвращает 32-битное число

Отсортируйте список по убыванию частоты появления элемента в нем.  
Если количество элементов равное, отсортируйте их по возрастанию.

Пример: `Solve([2,3,5,3,7,9,5,3,7]) => [3,3,3,5,5,7,7,2,9]`

Джонни - фермер, и он ежегодно проводит съезд фермеров, выращивающих свеклу, .

Каждый год он фотографирует рукопожатие фермеров. Джонни знает, что никакие два фермера не пожимают друг другу руки более одного раза. Он также знает, что некоторые возможные комбинации рукопожатий могут не состояться. Однако Джонни хотел бы узнать минимальное количество людей, принявших участие в этом году, просто посчитав все рукопожатия.

Помогите Джонни, написав функцию, которая принимает количество рукопожатий и возвращает минимальное количество людей, необходимое для выполнения этих рукопожатий (пара фермеров пожимают друг другу руки только один раз).