Ссылка на задачи: https://the-evening-code.com/posts/ten-javascript-exercises-with-arrays

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Решение:

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Сложность алгоритма: **O(N)**

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

**Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание**

**Изображение выглядит как текст, Шрифт, снимок экрана, линия

Автоматически созданное описание**

Сложность алгоритма: **O(N/2)**

Вывод:

• входные в функции параметры **не иммутабельны**;

• для реверса элементов массива без создания копии массива можно просто поменять местами отзеркаленные от центра массива элементы.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Решение: Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Сложность алгоритма: **O(N)**

Вывод:

• Array.prototype.filter(item => {}) возвращает массив, отфильтрованный функцией которая пробрасывается аргументом. Если функция на соответствующем элементе возвращает true, то такой элемент проходит фильтрацию и добавляется в возвращаемый массив.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Решение:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как Шрифт, линия, снимок экрана

Автоматически созданное описание

Сложность алгоритма: **O(N)**

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Решение

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

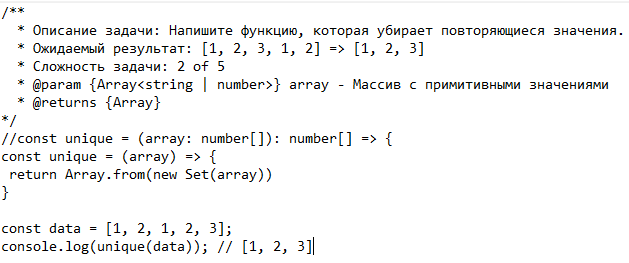


Сложность алгоритма: **O(N\*K)**

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Решение:



Изображение выглядит как текст, Шрифт, снимок экрана, линия

Автоматически созданное описание

Сложность алгоритма: **O(N)**

Вывод:

• Преобразовать объект Set в Array можно через Array.from(set);

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Решение:

Изображение выглядит как текст, снимок экрана, Шрифт, число

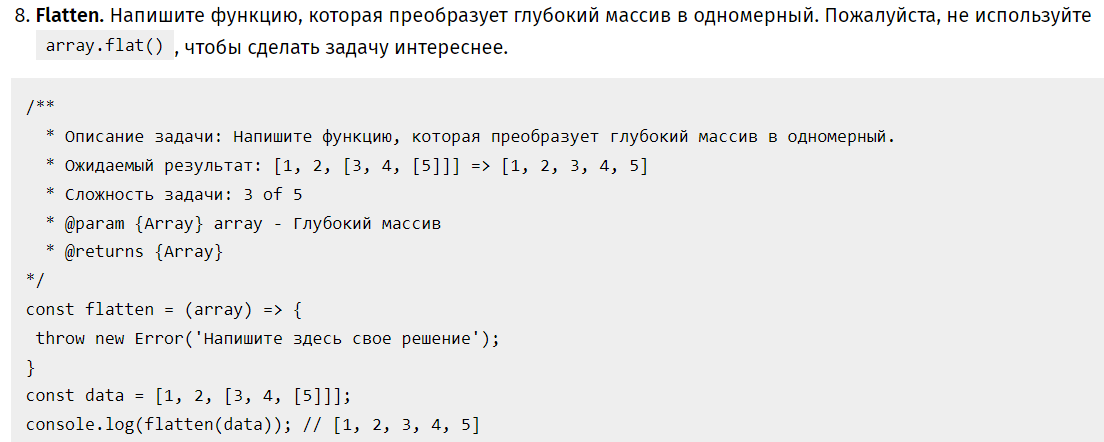
Автоматически созданное описание

Изображение выглядит как текст, Шрифт, линия, белый

Автоматически созданное описание

Сложность алгоритма: **O(N)**

Вывод:  
 • при сравнении двух массивов на эквивалентность элементов имеет смысл начать с проверки длин массивов. Если длины массивов разные => никак не могут быть равными.



Решение:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описаниеИзображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание

Асимптотическая сложность алгоритма: **???**

Вывод:

• рекурсия – частный случай замыкания:

Изображение выглядит как текст, рукописный текст, Прямоугольник, Параллельный

Автоматически созданное описание

• проверить является ли переменная массивом можно, используя Array.isArray(var);

• для того, чтобы поместить в массив элементы другого массива, можно воспользоваться оператором «spread».

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Решение:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Асимптотическая сложность алгоритма: **O(N/K) ~ O(N)**

Вывод:

• Для того, чтобы извлечь подмассив из массива используется Array.prototype.slice(from, to). Важно: [from, to). Также важно то, что этот метод **возвращает** подмассив по указанным индексам и **не изменяет** исходный массив.

• Аргументы Array.prototype.slice([from, to]) опциональны и можно вызывать метод без них. В таком случае: без аргументов вернётся массив целиком (можно использовать для копирования); с одним аргументом «start» вернётся подмассив исходного массива в диапазоне [start, array.length).

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Решение:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, Шрифт, число, линия

Автоматически созданное описание

Сложность алгоритма: O(N) + O(N2) + O(N) = O(2N + N2) ~ **O(N2)**

Вывод:

• исключить дубли из элементов массива можно через new Set(arr). А именно: пробросить массив в конструктор set и этот же set преобразовать в новый массив.