

4. Hard JS

- 1) Реализовать асинхронное выполнение трудной вычислительной задачи (без блокировки потока) Например, вычисление большого числа
- 2) Написать функцию-генератор, выдающую числа фибоначчи.
- 3) Сделать прокси, позволяющий использовать дробные значения в качестве индексов доступа к элементам массива (при этом должно возвращаться значение элемента с ближайшим целым индексом).
- 4) Реализуйте асинхронную загрузку 10-и изображений и выведите их в DOM в соответствующем порядке. Если при загрузке каких-то изображений произошла ошибка, то на месте этой картинки выведите сообщение об ошибке.
- 5) Написать функцию-генератор, выдающую треугольные числа.
- 6) Сделать прокси, позволяющий использовать объекты {name, id} в качестве индексов доступа к элементам массива (при этом должно возвращаться значение элемента с индексом равным значению поля id объекта).
- 7) Реализуйте асинхронную загрузку 10-и скриптов и выведите их в DOM в порядке возрастания времени загрузки файла. Если при загрузке какого-то скрипта произошла ошибка, то ни один скрипт добавлять в DOM не нужно. Вместо этого выведите сообщение об ошибке.
- 8) Написать функцию-генератор, выдающую простые числа.
- 9) Сделать прокси, позволяющий использовать массивы чисел в качестве индексов доступа к элементам массива (при этом должно возвращаться значение элемента с индексом равным максимальному значению массива).
- 10) Сделать прокси, позволяющий использовать строки в качестве индексов доступа к элементам массива (при этом должно возвращаться значение элемента с индексом равным количеству гласных в строке; строки содержат только латиницу, цифры и спецсимволы).