* Крестики-нолики на квадратном поле произвольного размера. Нужно написать функцию, которая делает ход за очередного игрока и проверяет, заканчивается ли игра на этом ходу. Конец игры — когда игрок заполнил строку/столбец или одну из двух диагоналей. Требуется сложность O(1).
* Реализовать malloc/free, которые выделяют память с заданным выравниванием. В решении можно использовать обычные malloc/free.
* Реализовать malloc/free в пределах 64КБ-массива.
* Реализовать циклический сдвиг массива на заданную величину.
* Определить, является ли один массив результатом циклического сдвига второго массива.
* Определить величину сдвига отсортированного массива.
* Перевернуть стек. Доступны операции pop, push, top, is\_empty. Использовать только O(1) явно выделяемой памяти, но можно использовать рекурсию.
* Выбрать K наибольших элементов из входного потока чисел.
* Определить медиану массива чисел. Массив размером 10ГБ, доступно памяти 2ГБ.
* Найти подпоследовательность с максимальной суммой (или произведением).
* Найти непрерывную подпоследовательность с максимальной суммой (или произведением).
* Найти подпоследовательность с суммой, максимально близкой к 0.
* Определить, есть ли у двух односвязных списков общая часть.
* Дан массив чисел. На вход подаются пары (a, b). Для каждой пары нужно вывести все числа массива, находящиеся между a и b, как можно более эффективным образом. Допускается предварительная обработка.
* В комнате N людей, один из которых знаменитость. Знаменитость характерна тем, что она никого не знает, но ее знают все. У нас есть матрица NxN. Элемент n(i,j) равен 1, если человек i знает человека j, и равен 0 в противном случае. Нужно за линейное время найти знаменитость или показать, что ее в комнате нет.
* Определить, что одна последовательность символов получена из другой путем некоторой перестановки символов. Пробелы и знаки препинания игнорировать.
* Найти N-ый наименьший элемент в массиве.
* Найти общие элементы двух массивов. Повторы не выводить.
* Реализовать quick sort.
* Реализовать merge sort.
* Реализовать двоичную кучу.
* Используя функцию compare-and-set (а-ля InterlockedCompareExchange), реализовать мьютекс.
* Реализовать очередь с синхронизированным доступом.
* Дано несколько потоков. Сделать так, чтобы они выполнялись в определенном порядке: поток 2 выполняется после завершения потока 1, поток 3 — после завершения потока 2, и т. д.
* Решить проблему читателей и писателей: есть область памяти, разделяемая несколькими потоками. Бывают потоки-читатели и потоки-писатели. К памяти может обращаться одновременно несколько читателей, но только один писатель. Читатели и писатели не могут обращаться к памяти одновременно.
* Проверить правильность скобочного выражения.
* Найти минимальное расстояние между двумя заданными символами в заданной строке.
* Инвертировать строку символов UTF-8.
* Изменить порядок слов в строке на противоположный.
* Реализовать сопоставление с образцом: дана строка и образец. Образец может включать специальные символы: '?' означает "любой символ", '\*' означает "любое количество любых символов". Определить, соответствует ли строка образцу.
* Отсортировать петабайт чисел.
* Есть много текстовых файлов. Пользователь вводит несколько слов. Нужно найти все файлы, каждый из которых содержит все введенные слова в любом порядке. Разработать необходимые структуры данных.
* Реализовать слияние отсортированных массивов.
* Реализовать слияние отсортированных односвязных списков.
* Даны два различных обхода BST. Построить третий.
* Переставить числа в массиве так, чтобы если их вывести подряд без пробелов, получилось бы наибольшее число.
* Два ровера высаживаются на одномерную планету на парашютах, которые остаются на месте посадки. Роверу доступны команды "влево", "вправо", "ничего не делать", "перейти на метку, если стою на парашюте". Написать программу поведения роверов, позволяющую им встретиться.
* Поменять местами биты на четных местах с битами на нечетных в 32-битном беззнаковом целом. 0-й бит меняется местами с 1-ым, 2-й с 3-им, и т. д.
* Подсчитать количество единичных битов в 32-битном беззнаковом целом.
* Поменять местами элементы на четных местах с элементами на нечетных местах в односвязном списке.
* Получить имя K-го столбца в Excel.
* Обход дерева "змейкой": по уровням сверху вниз, уровни обходятся то слева направо, то справа налево.
* Найти K-ый по величине элемент матрицы, в которой каждый столбец и каждая строка отсортированы по возрастанию.
* Найти заданное число в матрице, в которой каждый столбец и каждая строка отсортированы по возрастанию.
* Найти ближайшего общего предка двух узлов в BST.
* Найти в BST максимальный элемент на каждом уровне.
* Найти все общие элементы двух BST.
* Односвязный список содержит цикл. Найти узел, в котором начинается цикл, и размер цикла.
* В файле 4 млрд 32-битных беззнаковых чисел, каждое повторяется не более 1 раза. Найти все числа, отсутствующие в файле.
* Даны длины сторон треугольника. Определить его вид: обычный, равнобедренный, равносторонний или невозможный.
* Найти максимальную общую подстроку двух строк.
* Есть поток целых чисел. Более 50% последовательности составляет одно и то же число. Найти его с O(1) памяти.
* В BST найти элемент, следующий по порядку за заданным.
* Реализовать хэш-таблицу строк.
* Реализовать очередь, имея два стека.
* Реализовать стек, имея две очереди.
* Задан набор слов. Вывести все поднаборы, в каждом из которых все слова являются анаграммами друг друга.
* Найти первый символ строки, который не повторяется в ней два или более раз.
* Дан массив чисел. Найти в нем все тройки чисел a, b, c такие, что a^2 + b^2 = c^2. За время меньше O(N^2).
* Найти длиннейшую возрастающую подпоследовательность (не обязательно непрерывную).
* Получить все перестановки заданного массива.
* Есть генератор случайных чисел, с равной вероятностью выдающий числа от 1 до 5. Реализовать ГСЧ, равновероятно выдающий числа от 1 до 7.
* Перемешать числа в массиве. Все перестановки должны иметь равные вероятности.
* Случайным образом выбрать m чисел из n-элементного массива.
* Имеется набор слов. Даны два слова из этого набора. Нужно преобразовать первое слово во второе. На каждом шаге преобразования можно добавлять, удалять или изменять только одну букву. Каждое промежуточное слово также должно входить в набор.
* Даны два слова. Нужно преобразовать первое слово во второе. На каждом шаге преобразования можно добавлять, удалять или изменять только одну букву. Выполнить преобразование за наименьшее количество шагов.
* Найти подматрицу квадратной матрицы с наибольшей суммой элементов.
* Получить все k-элементные подмножества заданного множества.
* Имея набор достоинств монет, найти минимальное количество монет, требуемых для формирования заданной суммы (extra credit, если быстрое решение).

Еще задачи от Константина Логинова:

* В дереве (бинарном/не бинарном) все вершины - белые или черные. Найдите самый длинный полностью белый путь(корень может быть где угодно).
* Найти K-ую порядковую статистику в BST, когда а) у каждого узла есть ссылка на родителя или б) имеется только ссылка на корень дерева.
* Найти медиану в двух отсортированных массивов (одинакового/разного размеров).
* Найти наименьшее положительное число, пропущенное в отсортированном массиве.
* Найти дубликаты в массиве чисел (числа могут быть только от 0 до n-1).
* В отсортированном массиве все элементы имеют пары, за исключением одного. Найти его.
* Развернуть слова в строке (первое слово становится последним и т. д., сами слова сохраняются).
* Реализовать сопоставление с образцом: дан массив слов и образец. Образец может включать специальные символы: '?' означает "любой символ", '\*' означает "любое количество любых символов". Определить, соответствует ли какое-либо слово образцу.
* Распечатайте массив NxM по спирали.
* Напишите алгоритм, проверяющий, является ли дерево сбалансированным (разница высот любых 2-х поддеревьев узла отличается не более чем на 1).
* Отсортировать массив, сдвигая дубликаты в конец массива.
* Напишите реализацию функции IndexOf.
* Составьте все правильные скобочные выражения заданной длины (длина 4: "(())" и "()()").
* Написать программу определения количества шестизначных 'счастливых' билетов, у которых сумма первых 3 десятичных цифр равна сумме 3 последних десятичных цифр.
* У покупателя есть n монет достоинством H(1),..., H(n). У продавца есть m монет достоинством B(1),...,B(m). Может ли купить покупатель вещь стоимости S так, чтобы у продавца нашлась точная сдача (если она необходима).
* Разделить не отсортированный массив на 2, сумма элементов которых будет самой близкой.
* Распечатайте последние N строчек текстового файла.
* Передвинуть все нули в начало массива, не нарушая порядок следования ненулевых чисел (О(n)).
* Найти самый длинный палиндром в строке (за время О(n) с использованием памяти O(n)).
* Клонировать дерево, у которого слиплись некоторые подкорни в правильное дерево (на картинке нагляднее: <http://outcoldman.com/Library/2011/03/17/image_0168E3A9.png>).
* Дано n. Выведите числа от 0 до 2^n-1 (включительно) так, чтобы двоичное представление соседних чисел отличалось ровно на 1 бит.