1. Дан массив длины n, состоящий только из нулей и единиц. Предложите линейный алгоритм сортировки данного массива.

1) Заводим переменные счетчики кол-ва нулей $zcounter$.

2) При встрече очередного 0 записываем в ячейку с индексом $zcounter$ значение $'0'$ и инкрементируем счетчик.

3) Когда прошлись по всему массиву, все ячейки, начиная с $zcounter$, заполняем единицами.

2. На прямой задано n отрезков, причем известно, что они образуют систему строго вложенных отрезков (их можно упорядочить так, чтобы каждый строго содержался в следующем). Отрезки заданы координатами концов $[l_i, r_i]$ (и могут быть даны в неупорядоченном виде). Предложите асимптотически эффективный алгоритм (с точки зрения количества арифметических операций), который находит все точки прямой, которые покрыты ровно $2n/3$ отрезками.

1) Находим порядковую статистику с номерами $\frac{2n}{3}$ и $\frac{2n}{3}+1$. Поиск такой порядковой статистики пройдет за линейное время

2) Найденные промежутки (с выколотыми ближе к центру точками) по сути и будут искомыми, то есть мы решили задачу за линейное время!

3. Рассмотрим детерминированный алгоритм поиска порядковой статистики за линейное время из параграфа 9.3 Кормена. Какая асимптотика будет у алгоритма, если делить элементы массива на группы по семь, а не по пять?

1) Разбиваем массив на блоки по 7 – также за линейное время

2) Чтобы найти медиану в массиве медиан, нам понадобится $T(\frac{n}{7})$ операций

3) Далее определяем место, где будет находиться медиана медиан: $\frac{2n}{7} \leqslant med \leqslant \frac{5n}{7}$

4) В таком случае сложность алгоритма будет похожа на ту, что была получена в википедии: $T(n) = T\frac{5n}{7} + T(\frac{n}{7}) + cn$. Тогда $T(n) = \Theta(n)$

4. На вход задачи подаётся число n и массив чисел $x_1, x_2, \ldots, x_{2n+1}$. Постройте линейный алгоритм, находящий число s, при котором достигается минимум суммы

$$\sum_{i=1}^{2n+1} |x_i - s|.$$

Чтобы найти это число s, необходимо найти "центр тяжести"всех точек, расположенных на прямой с координатами x_i.

В одной из задач выше мы находили медиану за линейное время. Здесь по сути нам тоже нужно воспользоваться алгоритмом поиска порядковой статистики за линейное время.

С помощью этого алгоритма мы найдем медиану, которая и будет являться искомой s.

5. Предложите полиномиальный от длины входа алгоритм решения сравнения $a \cdot x + b \equiv 0 \pmod{M}$ (На вход дают целые числа a, b, M в двоичной системе исчисления).

Раньше мы сталкивались с похожими уравнениями, поэтому знаем, что $ax + b \equiv 0 (mod M)$ можно представить в виде $ax + b = M \cdot k$, где $k \in Z$ – какое-то целое число.

1	0	3	2	0	0	0	0
1	3	0	0	0	2	0	0
1	0	3	0	0	0	2	0
1	1	3	3	0	0	2	2
4	1+3i	-2	1-3i	2	2i	-2	-2i
6	1+3i+$\sqrt{2}$(i-1)	-2-2i	1-3i-$\sqrt{2}$(i-1)	2	1+3i-$\sqrt{2}$(i-1)	-2+2i	1-3i+$\sqrt{2}$(i-1)

Таблица 1: Быстрое преобразование Фурье

С помощью расширенного алгоритма Евклида находим решения, которые будут равны $x = x_0 + \frac{M}{\textbf{НОД}} q, q \in Z$.

Таким образом, сложность алгоритма будет зависеть исключительно от алгоритма Евклида, который работает за полиномиальное время.

6. Перемножьте многочлены $2x^3+3x^2+1$ и $2x^2+x$ с помощью БПФ. В решении должны быть приведены вычисления всех используемых преобразований.

$A(x) = 2x^3 + 3x^2 + 1$
$B(x) = 2x^2 + x$
$A(x)B(x) = P_5(n)$

7. Решите с помощью преобразования Фурье задачу о поиске всех вхождений образца с джокерами в текст. Текст и образец — это последовательности $t_0, t_1, \ldots, t_{n-1}$ и $p_0, p_1, \ldots, p_{m-1}$, $m < n$, где все t_i — символы из алфавита, а p_j — либо символ из алфавита, либо джокер. Образец входит в текст в позиции $i \in \{0, \ldots, n-m-1\}$, если $t_{i+j} = p_j$ при всех $j \in \{0, \ldots, m-1\}$, для которых p_j — символ алфавита. Для решения этой (и более сложной задачи в домашнем задании) есть $O(n \log n)$ алгоритм, основанный на БПФ. Закодируем каждый символ алфавита уникальным положительным числом, а джокер нулём, и определим последовательность r_i: $r_i = \sum_{j=0}^{m-1} p_j t_{i+j} \left(p_j - t_{i+j}\right)^2$

1. Докажите, что образец входит в текст в позиции i тогда и только тогда, когда $r_i = 0$.

2. Постройте $O(n \log n)$ алгоритм, который находит все вхождения образца с джокерами в текст.

Заметка. Эта задача подготовлена на основе статьи P. Clifford, R. Clifford Simple deterministic wildcard matching, Information Processing Letters, Vol. 101, Is. 2, 2007, Pp. 53-54,

8 [ДПВ 2.30]. В данном упражнении показывается, как вычислять преобразование Фурье (ПФ) в арифметике сравнений, например, по модулю 7.

1. Существует такое ω, что все степени $\omega, \omega^2, \ldots, \omega^6$ различны (по модулю 7). Найдите такое ω и покажите, что $\omega + \omega^2 + \ldots + \omega^6 = 0$. (Отметим также, что такое число существует для любого простого модуля.)

Эта ω равна 3, т.к.
$3^6 = 729 \equiv 1 mod 7$
$3^2 = 9 \equiv 2 mod 7$
$3^1 = 3 \equiv 3 mod 7$
$3^4 = 81 \equiv 4 mod 7$
$3^5 = 243 \equiv 5 mod 7$

$3^3 = 27 \equiv 6 mod 7$

$\omega + \omega^2 + \cdots + \omega^6 \equiv 0 mod 7$ – это следует непосредственно из того, что сумма остатков (21) делится на 7.

2. Найдите преобразование Фурье вектора $(0, 1, 1, 1, 5, 2)$ по модулю 7, используя матричное представление, то есть умножьте данный вектор на $M_6(\omega)$ (для найденного ранее ω). Все промежуточные вычисления производите по модулю 7.

Это будет

3. **Запишите матрицу обратного преобразования Фурье. Покажите, что при умножении на эту матрицу получается исходный вектор. (Как и прежде, все вычисления должны производиться по модулю 7.)**

4. **Перемножьте многочлены $x^2 + x + 1$ и $x^3 + 2x - 1$ при помощи ПФ по модулю 7.**