

1. С помощью сортировки слиянием определите количество инверсий в перестановке. Перестановкой длины n называется массив из n попарно различных целых чисел от 1 до n . Пара чисел a_i и a_j образуют инверсию, если $i < j$, но $a_i > a_j$.
2. Пусть в алгоритме быстрой сортировки в качестве пивота всегда детерминированно выбирается центральный элемент массива. Для произвольного n за $O(n)$ постройте перестановку, на котором такая сортировка занимает $\Omega(n^2)$ времени.
3. Пусть некоторая перестановка задана в виде массива.
 - а) За одну операцию можно поменять местами два соседних элемента. Докажите, что минимальное количество операций, необходимое для сортировки массива, равно числу инверсий.
 - б) За одну операцию можно поменять местами два любых элемента. Докажите, что чётность числа инверсий изменяется после каждой операции.
4. На прямой заданы n отрезков координатами своих концов $[a_i, b_i]$. Найдите
 - а) длину их объединения (асимптотика: $O(n \log n)$);
 - б) длину их пересечения (асимптотика: $O(n)$);
 - в) максимальное количество отрезков, которое можно выбрать так, чтобы выбранные отрезки попарно не пересекались (асимптотика: $O(n \log n)$).
5. Напомним, что процедура $\text{Partition}(A, x)$ переупорядочивает элементы массива A так, что сначала идут все элементы, не превосходящие x , в некотором порядке, а затем — все элементы, большие x . Покажите, как реализовать $\text{Partition}(A, x)$ с привлечением $O(1)$ дополнительной памяти.
6. Пусть A — массив длины n , а B — его отсортированная версия. Найдите за $O(n \log n)$ перестановку σ , такую что $b_i = a_{\sigma(i)}$ для всех i . В массиве A могут быть повторяющиеся элементы.

1. Достаточно по двум отсортированным массивам a_1, \dots, a_m и b_1, \dots, b_k найти количество пар чисел i и j , для которых $a_i > b_j$.
2. Какой элемент должен быть центральным, чтобы длина сортируемых кусков была наименее оптимальной?
3.
 - а) Используйте (и докажите), что если в массиве число инверсий положительно, то обязательно есть два соседних элемента, образующих инверсию.
 - б) Определите, остаётся ли (становится ли) пара (i, j) инверсией после обмена местами чисел на позициях k и l ?
4.
 - а) Воспользуйтесь методом сканирующей прямой: отсортируйте все отрезки по возрастанию левых концов и пройдите его слева направо.
 - б) Это задача-шутка.
 - в) Отсортируйте отрезки по возрастанию правых концов. Возьмите первый из них. Какой можно взять следующим? Почему такой жадный алгоритм даёт оптимальный ответ?
- 5.

```
int i = 0, j = n - 1;
do {
    while (a[i] < p) ++i;
    while (a[j] > p) --j;
    if (i <= j) {
        std::swap(a[i], a[j]);
        ++i;
        --j;
    }
} while (i <= j);
```

6. Для каждого i можно запустить бинарный поиск, использованные числа в B следует помечать.