

Пронумеруем все начала отрезков a_i в порядке возрастания координаты. Если где-то координаты совпадают, порядок нумерации определяется по убыванию координат концов, если же и координаты концов совпадают — произвольным образом. Далее пронумеруем концы отрезков так, чтобы номер конца отрезка совпадал с номером его начала. Выпишем получившиеся номера концов отрезков в порядке убывания их координаты. Если где-то координаты совпадают — выписываем их в произвольном порядке. Сложность — $O(n \log n)$.

В результате получилась некоторая перестановка σ чисел от 1 до n . Заметим, что индекс вложенности i -го отрезка равен количеству чисел, которые в этой перестановке одновременно

- (а) находятся левее (меньше по индексу) числа i
- (б) меньше (по значению) числа i

Исключение составляют отрезки, которые полностью совпадают, но мы можем не обращать на это внимание, поскольку у одного из таких отрезков результат будет правильным (а у остальных — меньше), а следовательно, это не меняет ответ на вопрос задачи (считаем, что если отрезки совпадают, то каждый из них содержит все остальные).

По перестановке σ мы можем найти массив β чисел, в котором число β_i равно количеству таких чисел в перестановке σ , что они левее (меньше по индексу) и меньше (по значению) числа σ_i .

Алгоритм нахождения β по σ практически идентичен классическому алгоритму нахождения таблицы инверсий по перестановке. Положим $\beta_i = 0$, $i = 1 \dots n$. Пусть k пробегает значения от $\lfloor \lg n \rfloor$ до 0 (иными словами, k пробегает количество цифр в двоичной записи максимального числа в σ). Для каждого k будем обнулять все счетчики $x_s : 0 \leq s \leq n/2^{k+1}$ и для всех $j = 1, 2, \dots, n$ выполнять следующее: вычисляем $r = \lfloor \sigma_j / 2^k \rfloor \bmod 2$, $s = \lfloor \sigma_j / 2^{k+1} \rfloor$; если $r = 0$, то $x_s += 1$, если же $r = 1$, то $b_j += x_s$.

По существу алгоритм делает следующее. Запишем все числа перестановки в двоичной системе и дополним слева нулями так, чтобы длина всех записей была одинакова. Тогда на шаге $k = \lfloor \lg n \rfloor$ мы выписываем количество таких чисел в перестановке σ , что они меньше по индексу, а их старший бит меньше старшего бита числа σ_i . На шаге $k = \lfloor \lg n \rfloor - 1$ мы решаем задачу уже для двух старших битов и так далее. Сложность — $O(n \log n)$. В итоге нам нужно проверить, есть ли в массиве β число, большее чем 1000, что делается за $O(n)$. Итоговая сложность — $O(n \log n)$.