Разбор задачи «Ребрендинг»

Ниже будет приведено два решения, которые проходили все тесты.

Решение за $\mathcal{O}(n\sqrt{n} + q \log q)$:

Пусть длина отрезка из запроса равна $\mathcal{O}(k)$. Тогда ответ не превосходит $\mathcal{O}(\frac{n}{k})$. Действительно, пусть ответ равен $x > \mathcal{O}(\frac{n}{k})$, тогда посмотрим на худший случай. Будем расставлять числа жадно – $1, x+1, 2x+2, \ldots, n$. Всего чисел будет $\mathcal{O}(\frac{n}{x}) < \mathcal{O}(\frac{n}{n/k}) = \mathcal{O}(k)$. Значит чисел будет меньше чем мы предполагали – получили противоречие. Таким образом получаем утверждение, что если длина отрезка $\geqslant \mathcal{O}(k)$, то ответ $\leqslant \mathcal{O}(\frac{n}{k})$.

Давайте возьмем $k = \sqrt{n}$. Для всех отрезков длины $\leqslant k$ посчитаем их ответ за $\mathcal{O}(k \log k)$ или $\mathcal{O}(k)$ (первое делается сортировкой всего отрезка). Для остальных отрезков нас будут интересовать только такие i, j, что $|a_i - a_j| \leqslant \mathcal{O}(\sqrt{n})$. В зависимости от реализации обработки всех этих пар и релаксированние для них ответов ваше решение могло получать от 62 до 90 баллов.

Решим сначала задачу для запросов с длиной отрезка $\leqslant \sqrt{n}$. Будем идти по элементам слева направо, и для каждого элемента поддерживать $dp[i]$ – минимальная разность элемента i с элементами правее, которые мы успели рассмотреть. Более формально, пусть r – текущая правая граница, тогда $dp[i] = \min\limits_{i<j\leqslant r} |a[i] - a[j]|$. При переходе к $r+1$ нам нужно пересчитать $dp[i]$ только для $r+1-\sqrt{n} \leqslant i \leqslant r$. Используя $dp[i]$ можно легко отвечать на запросы, длина которых $\leqslant \sqrt{n}$ Действительно, давайте для i отрезка, когда мы будем в позиции l_i посмотрим на $\min\limits_{l_i \leqslant j < r} dp[j]$. Это и будет ответом. Эта часть решения работает за $\mathcal{O}(n\sqrt{n})$.

Давайте теперь обработаем отрезки, длина которых $\geqslant \sqrt{n}$. В этом случае ответ $\leqslant \sqrt{n}$. Для каждого числа есть только $2\sqrt{n}$ чисел, которые образуют с ним разницу $\leqslant \sqrt{n}$. Давайте будем идти точно также слева направо и для каждого ответа от 1 до $\sqrt{n}$ поддерживать максимальную левую границу, при которой мы можем набрать эту разницу. Обозначим это за $dp2[i]$. Пусть мы посчитали $dp2[i]$ для r, перейдем к $r+1$. Нам нужно обработать все пары, в которые входит $a[r+1]$ и разность в которых меньше $\sqrt{n}$. Это можно сделать явно суммарно за $\mathcal{O}(n\sqrt{n})$. Для каждого отрезка мы можем явно найти первый ответ ans, для которого $dp2[ans] \leqslant l_i$. Но мы несложно заметить, что если $l_{i-1} \leqslant l_i$, то $ans_{i-1} \leqslant ans_i$. Мы можем двигать указатель и получить ответ для всех отрезков такого типа за $\mathcal{O}(n\sqrt{n} + q \log q)$ (потому что нужно еще посортировать отрезки). При должном старании и подборе k это решение проходило все тесты.

Решение за $\mathcal{O}(n \log^2 n + q \log n)$:

Давайте будем также идти по всем элементам слева направо. Основной задачей будет поддержка актуальной версии $dp[i]$ – минимальной разности a_i с элементами справа от него, который мы успели рассмотреть. Пусть мы корректно посчитали dp для первых r элементов. Перейдем к $r+1$. Давайте покажем как обновить ответ для всех $j < i$, таких что $a[j] > a[i]$. Для $j < i$, таких что $a[j] < a[i]$ решается аналогично.

Давайте возьмем первый элемент $a[j]$ слева от i, такой что $a[j] > a[i]$. Заметим, что если есть $l < j < i$, такой что $a[l] > a[j] > a[i]$, то для него мы $dp[l]$ не будем обновлять, потому что $|a[l] - a[j]| < |a[l] - a[i]|$. Также мы не будем обновлять ответ для l таких что $|a[l] - a[j]| < |a[l] - a[i]| \to a[l] > a[i] + \frac{a[j]-a[i]}{2}$. То есть дальше нас будут интересовать только числа с отрезка $\left[a[i], a[i] + \frac{a[j]-a[l]}{2}\right]$.

Давайте заметим, что мы уменьшили длину отрезка в 2 раза. То есть таких иттераций будет не больше $\mathcal{O}(\log n)$. Находить самое правое число, принадлежащее отрезку, можно с помощью дерева отрезков. Ответом для отрезка l_i, r_i будет $\min\limits_{l_i \leqslant j < r} dp[l]$ в момент r_i. Это также можно эффективно находить с помощью дерева отрезков. Итоговая асимптотика решения $\mathcal{O}(n \log^2 n + q \log n)$.