

Чудеса природы

Автор задачи: Егор Юлин, разработчик: Константин Бац

Для решения этой задачи вспомним, как выглядит динамика для поиска наибольшей общей подпоследовательности:

$$dp_i = \max_{j < i, a_j < a_i} dp_j + 1,$$

где dp_i соответствует длине НВП, заканчивающейся в i -м элементе.

От нас же по задаче требуется найти, во-первых, не длину максимальной подпоследовательности, а их количество (стандартное изменение пересчета динамики с $\max(\dots)$ на $\text{sum}(\dots)$), а во-вторых, не просто возрастающих последовательностей, а сначала возрастающих, затем убывающих.

Стандартная идея, применимая в таких ситуациях — разбить искомые последовательности по «центральному» элементу. Заметим, что если перебирать a_i , слева от которого подпоследовательность возрастает, а справа — убывает, то ответ на задачу равен

$$\sum_{i=1}^n \#lis_i \cdot \#lds_i,$$

где $\#lis_i$ — число возрастающих подпоследовательностей, заканчивающихся в a_i , а $\#lds_i$ — число убывающих подпоследовательностей, начинающихся в a_i .

Более того, заметим, что если мы научимся считать $\#lis$, то $\#lds$ можно просто посчитать как $\#lis$ на развернутом массиве a .

Ну а для подсчета $\#lis$ вспомним решение задачи о поиске НВП через дерево отрезков: будем обрабатывать элементы массива по очереди слева направо, а значения динамики dp_i хранить по индексу a_i . Тогда все элементы с $i < j$ и $a_i < a_j$ — это просто префикс нашего дерева отрезков. Поскольку мы хотим находить число подпоследовательностей, а не максимальную из них, пересчет будет выглядеть как $dp_i = \sum_{x < a_i} \text{segment_tree}_x$, то есть как запрос к ДО на префиксе.

Всего на подсчет $\#lis$ и $\#lds$ уйдет $\mathcal{O}(n \log n)$ времени, после чего ответ можно посчитать за линейное время, перебрав центральный элемент холма (см. формулу выше).