

---

## Разбор задачи «Не так грубо!»

Первая подзадача может быть решена простым разбором случаев.

Вторая подзадача решается простой реализацией того, что описано в условии: переберем границы выбранной подстроки и проверим, что она подходит, перебрав все пары символов в ней. Из всех подходящих подстрок выберем самую длинную. Асимптотика времени работы решения составляет  $O(n^4)$ .

Научимся считать количество пар символов, первый из которых равен «a», а второй — «b», за линейное время. Для этого переберем символ строки, и если он равен «b», то прибавим к ответу количество символов, равных «a», стоящих до него. Заметим, что эту величину можно не считать каждый раз заново, а последовательно идти по символам строки, храня количество пройденных символов «a», и если очередной символ равен «b», то прибавлять к ответу это количество.

Используя этот метод для проверки каждой из подстрок, получаем решение за  $O(n^3)$ .

Чтобы еще ускорить это решение, заметим, что если зафиксировать начало подстроки и перебирать ее конец в порядке возрастания индекса, то не нужно каждый раз заново запускать алгоритм подсчета количества искомых пар символов, ведь необходимые значения для подстроки  $s_l s_{l+1} \dots s_r$  можно получить из значений для подстроки  $s_l s_{l+1} \dots s_{r-1}$  так же, как это делается в алгоритме. Асимптотика времени работы решения —  $O(n^2)$ .

Для того, чтобы получить полное решение задачи, воспользуемся методом двух указателей. Для каждого левого конца подстроки найдем максимальный из правых концов, для которых соответствующая подстрока имеет грубость не более  $c$ . Заметим, что при увеличении индекса левой границы индекс правой границы не будет уменьшаться. Таким образом, мы можем поддерживать текущие левую и правую границы, сдвигать левую на один символ вправо, а далее расширять подстроку, пока это возможно. Для этого нам надо научиться пересчитывать количество искомых пар символов при добавлении символа в конец строки и удалении символа из начала строки.

Давайте модифицируем для этого алгоритм нахождения количества искомых пар, описанный выше. Будем хранить количество искомых пар, количество символов «a», которое содержит текущая подстрока, а также количество символов «b», которое она содержит. Тогда при добавлении символа в конец строки, если это символ «b», надо увеличить количество искомых пар на количество символов «a», а при удалении, если это символ «a», надо уменьшить количество искомых пар на количество символов «a».

Асимптотика данного решения —  $O(n)$ .