

Плагиат кода

Автор задачи и разработчик: Владислав Власов

Заметим, что порядок удаления пар соседних символов не важен. При любом порядке удаления из s будут удалены несколько отрезков подряд идущих символов, каждый из которых будет четной длины. Такого же результата всегда можно добиться, удаляя пары стоящих рядом символов по очереди слева-направо.

Так же сразу заметим, что если длины s и t имеют разную четность, то ответ всегда «NO», потому что удаление двух символов сохраняет четность длины строки.

Для прохождения **первой подгруппы** достаточно написать полный перебор. Будем обрабатывать символы строки s по очереди и для каждого перебирать, удалить его вместе со следующим или нет. Решение работает за $\mathcal{O}(2^{|s|})$, что с запасом проходит по времени.

Вторая подгруппа требовала анализа строки s . Если в t есть только символы 'a', то все символы s , не равные 'a', должны быть удалены. Как уже было сказано выше, удалить можно только четные отрезки подряд идущих символов, поэтому достаточно было проверить, что при удалении всех отрезков из «лишних» символов строка t не станет слишком короткой.

Чтобы получить конечную длину строки после удаления всего лишнего, достаточно найти все отрезки «лишних символов», что можно сделать за время $\mathcal{O}(|s|)$. Пройдемся по строке s , поддерживая счетчик символов, не равных 'a'. Когда встречается очередной символ 'a', очередной отрезок закончился, и надо запомнить значение счетчика, после чего обнулить его. Теперь, каждый четный отрезок можно удалить сам по себе, а каждый нечетный — только захватив стоящую рядом 'a'.

Надо также учесть, что если строка начинается и заканчивается символом, не равным 'a', все отрезки нечетны, и между ними стоит по одному символу 'a', то в принципе невозможно все эти отрезки удалить.

В **третьей подгруппе** не требовался подробный анализ строки s . Если гарантируется, что $|s| = |t| + 2$, то из s надо удалить ровно два символа. Можно перебрать позицию удаляемых символов, после чего сравнить полученную строку и t за длину строки. Такое решение работает за $\mathcal{O}(|s|^2)$.

Полное решение опирается только на идею удаления символов слева-направо. Посмотрим на первые символы s и t . Если они совпадают, то нет необходимости удалять первый символ из s . Докажем это: пусть в правильном ответе на самом деле s_1 удаляется вместе с s_2 , и, возможно, следующими символами, и на первом месте стоит s_i . Тогда i — нечетное, потому что было удалено четное число символов. Но тогда на самом деле можно было удалить все символы с s_2 по s_i включительно, и строка бы не изменилась, так как $s_1 = t_1 = s_i$.

Осталось только жадно удалить все символы, которые не совпадают с соответствующим символом t . Будем идти по строке s , и, встречая $s_i \neq t_i$, будем удалять s_i вместе с s_{i+1} . Действительно, мы доказали, что если $s_i = t_i$, то его можно оставить на месте, тогда как, очевидно, если $s_i \neq t_i$, а все предыдущие символы уже совпали, s_i придется удалить.

Для полного решения надо было реализовать описанную выше идею без явного удаления символов. Удалять символы из середины строки можно только за время, пропорциональное ее длине. Чтобы получить время $\mathcal{O}(|s|)$, будем поддерживать указатель на следующий неудаленный символ s . Если он указывает на символ, равный t_i , переходим к следующим. Иначе «удаляем» два символа из s , то есть просто двигаем указатель на две позиции вперед. Если указатель дойдет до конца s раньше, чем найдется совпадение со всей t , ответ — «NO», иначе (в случае, если $|s| \bmod 2 = |t| \bmod 2$) — «YES».