Мысленно запомним, какие пары индексы в течение работы алгоритма стали соседними командами. На пример, для строки ‘bdcef’ такими парами были $(1, 2), (0, 2), (3, 4)$. Тогда можем заметить, что эти пары отсортированы по второму элементу в паре. Это верно, так как по условию всегда удаляется самая левая пара, и если после текущей пары существует пара с меньшим вторым индексом, то могли и до этого ее обработать. Поэтому получается следующая идея алгоритма.

Будем перебирать второй индекс в паре от 0 до $n - 1$ и будем поддерживать текущий префикс строки после удалений. Тогда текущий индекс может быть в паре только с последним символом строки. Если они не образуют пару, то добавим текущий символ в конец строки и перейдем к следующему символу. Если текущий символ это меньший символ в паре, то должны его удалить, текущая строка не меняется, и переходим к следующему символу. Иначе удалим последний символ строки и снова рассмотрим случаи уже с новым последним символом.

Реализовать данный алгоритм можем с помощью стека. Итоговая сложность будет $O(n)$, так как обрабатываем каждый символ не более одного раза и удаляем каждый символ тоже не более одного раза.