

Compression (dynamic programming)

The problem must be solved using a dynamic programming (DP) approach, and the time complexity should be less than or equal to $O(N^3)$.

A worker has developed a new, great data compression strategy, called “Don’t Repeat Yourself” (DRY). DRY works on a string of characters, and if the string contains two consecutive instances of the same substring, it simply removes one of them. For example, in the string “1010”, it could remove one of the two “10” substrings, resulting in “10”.

If you try to compress '101', since there are no consecutive identical substrings in the string, the result will remain as '101'.

DRY can also take advantage of previous removals — for instance, in the string “10110”, it will first remove one of the consecutive '1's in the 3rd and 4th positions, resulting in “1010”, and then remove the duplicate “10” in “1010”, resulting in “10”.

If there are multiple choices of repeating consecutive substrings to remove, DRY should choose in a way that results in the shortest possible final string.

For all binary strings, you should output the shortest string that can be obtained by repeatedly applying DRY to it.

Let’s look at a few more examples. When '101100010' is given as input, there are several ways to compress it. Among them, one of the compression sequence that yields the shortest result is as follows. First, compress the '00' in the 5th and 6th positions of '101100010', which results in '10110010'. Next, compress the '00' in the 5th and 6th positions of '10110010', resulting in '1011010'. Then, by compressing the last four characters '1010', you get '10110'. Compressing the '11' in the 3rd and 4th positions of '10110' gives '1010'. Finally, compressing '1010' results in '10'.

When the input is '1010011100'. First, remove the duplicate '10' from the first four characters '1010' to get '10011100'. Then, compress the '11' at the 4th and 5th positions to obtain '1001100'. Again, compress the '11' at the 4th and 5th positions to get '100100'. From '100100', remove the duplicate '100' to get '100' as the result. Finally, remove one of the duplicate '0's at the end to get '10'.

The type of Input and Output is Str. You must solve this question with DP.

The length of Input n , $1 \leq n \leq 100$

Input / Output

1010	10
101	101
10110	10
101100010	10
1010011100	10