

S and T are strings composed of lowercase letters. In S, no letter occurs more than once.

S was sorted in some custom order previously. We want to permute the characters of T so that they match the order that S was sorted. More specifically, if x occurs before y in S, then x should occur before y in the returned string.

Return any permutation of T (as a string) that satisfies this property.

Example :Input: S = "cba" T = "abcd" Output: "cbad" Explanation: "a", "b", "c" appear in S, so the order of "a", "b", "c" should be "c", "b", and "a". Since "d" does not appear in S, it can be at any position in T. "dcba", "cdba", "cbda" are also valid outputs.

Note:

- S has length at most 26, and no character is repeated in S.
- T has length at most 200.
- S and T consist of lowercase letters only.