Problem J. Suffix

Consider n given non-empty strings denoted by $s_1, s_2, \cdots, s_n$. Now for each of them, you need to select a corresponding suffix, denoted by $suf_1, suf_2, \cdots, suf_n$. For each string s_i, the suffix suf_i is a non-empty substring whose right endpoint is the endpoint of the entire string. For instance, all suffixes of the string "jiangsu" are "u", "su", "gsu", "ngsu", "angsu", "iangsu" and itself.

All selected suffixes could assemble into a long string $T = suf_1 + suf_2 + \cdots + suf_n$. Here plus signs indicate additions of strings placing the latter at the tail of the former. Your selections of suffixes would determine the lexicographical order of T. Now, your mission is to find the one with minimum lexicographical order.

Here is a hint about lexicographical order. To compare strings of different lengths, the shorter string is usually padded at the end with enough "blanks" which is a special symbol that is treated as smaller than every letters.

Input

The first line of input contains an integer T which is the total number of test cases. For each case, the first line contains an positive integer n. Each of the following n lines contains a string entirely in lowercase, corresponding to $s_1, s_2, \cdots, s_n$. The summation of lengths of all strings in input is smaller or equal to 500000.

Output

For each test case, output the string T with minimum lexicographical order.

Sample

3 3 bbb aaa ccc 3 aba aab bab 2 abababbaabbababba abbabbabbbababbab	baaac aaabab aab

This page intentionally left blank.