

Edit Distance

The edit distance of two strings S and T is the minimum number of edit operations that need to be done to transform S into T . The valid edit operations are:

- Insert a single character at any position.
- Modify an existing character.
- Remove an existing character.

For example, the edit distance of “pantera” and “aorta” is 5, because the following chain of edits is valid (and there is no shorter chain):

“pantera” \ggg “antera” \ggg “aotera” \ggg “aoera” \ggg “aora” \ggg “aorta”.

In this problem, given a value K and a word S , we need to construct a word T such that the edit distance of S and T is at most K . There are of course several possibilities for that, so we will ask that you choose the word T that comes first alphabetically. A word always comes alphabetically after any proper prefix. Among two words that are not prefixes of each other, the one that comes first alphabetically is the one that has, in the first position at which they differ from left to right, a letter closest to the beginning of the alphabet. Notice that the empty word (that has zero characters) is a valid word and is alphabetically before any other word.

Input

The input contains several test cases. Each test case is described in a single line that contains an integer K ($0 \leq K \leq 1000$) and a non-empty word S of at most 1200 lowercase letters, separated by a single space. The last line of the input contains the number -1 and an asterisk separated by a single space and should not be processed as a test case.

Output

For each test case output a single line with a word T of lowercase letters such that the edit distance of S and T is at most K . If there are several words in that situation, output the one that comes first alphabetically.

Example

Input:

```
4 abcdef
1000 zero
0 zero
-1 *
```

Output:

```
aaaaaabcdef
```

```
zero
```