

Problem G. String Transformation

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 256 mebibytes

Cody wrote down a string of English letters and an integer k . Now he wants to transform this string so that it would have exactly k “holes”. A hole is a closed loop in a letter. For example, the letter “B” has two holes, and the letter “a” has one hole.

Cody can transform the string by choosing a letter and “incrementing” it, changing it into the next letter of the alphabet (for the last letter of the alphabet, the next one is the first letter of the alphabet). Cody can repeat this action as much as he wants with any letter in the string.

In Cody’s understanding, the alphabet is this sequence of English letters:

AaBbCcDdEeFfGgHhIiJjKkLlMmNnOoPpQqRrSsTtUuVvWwXxYyZz

All the letters Cody writes look exactly like the letters in the alphabet depicted above.

Naturally, Cody wants to perform as little increments as possible to achieve his goal. If the string cannot be transformed to contain the needed amount of “holes”, Cody sees it and just doesn’t bother to perform any transformations. Can you be as smart as Cody?

Input

The first line of input consists of two space-separated integers n and k : the length of the string and the required number of “holes” ($1 \leq n \leq 10^5$, $-10^{18} \leq k \leq 10^{18}$).

The second line is the string s Cody wrote down. The string s includes only uppercase and lowercase English letters.

Output

Output the number of increments needed to achieve the goal, or -1 in case it is impossible.

If the string can be transformed, the second line of output should be the resulting string. If there are several possible answers, print any one of them.

Example

standard input	standard output
5 3 z1Ena	2 zlenB