Problem A. Alphabet City

Time limit: 3 seconds
Memory limit: 1024 megabytes

Al is an urban designer in Alphabet City, and today their task is to prepare signs for n city streets. In Alphabet City, the street signs simply consist of the street names composed of capital identically stamped English metal letters. For instance, if, during nighttime, someone sneakily exchanges the first letters on NERC street and on NEF street, the next day nobody will see the difference as the letter 'N' is identical on both signs.

Al is planning to put m signs on each street, and they have already ordered the required number of brass letters for each of the street names from the metallurgical plant. However, one hour before the order arrived, Al got a call from the plant's manager with a devastating piece of news: they lost one of the items from the list of street names! To mitigate the issue, Al decided for now to put as many signs as possible on each street whose order was not lost, and, with the leftover letters, to prepare at least one sign for the street whose order was lost.

Formally, if $s_1, \ldots, s_n$ are the street names, and ℓ is the index of the missing item from the order, Al is interested in the maximum integer k such that it is possible, from all the letters of m copies of $s_1, \ldots, s_{\ell-1}, s_{\ell+1}, \ldots, s_n$, to compose k copies of $s_1, \ldots, s_{\ell-1}, s_{\ell+1}, \ldots, s_n$ and additionally at least one copy of s_ℓ, or to determine that such a composition is impossible for any non-negative k.

Al still does not know which of the items was lost. Write a program that, given m and all street names, for each $\ell \in \{1, 2, \ldots, n\}$ prints the maximum value of k, or -1 if such a composition is impossible.

Input

The first line consists of two integers n and m, denoting the number of streets in Alphabet City for which the signs are needed and the number of copies of each street name that Al initially ordered ($2 \le n \le 2 \cdot 10^5$; $1 \le m \le 5 \cdot 10^5$). Each of the next n lines consists of one string s_i — the street name ($1 \le |s_i| \le 5 \cdot 10^5$). All these names are composed of capital English letters. Some or all of these names *may* coincide. It is guaranteed that the sum of the lengths of all the street names does not exceed $5 \cdot 10^5$.

Output

Print n integers, the ℓ-th of them denoting the maximum integer k such that the letters of $m \times s_1$, …, $m \times s_{\ell-1}$, $m \times s_{\ell+1}$, …, $m \times s_n$ (where $m \times s$ denotes m copies of street name s) are enough to compose $k \times s_1$, …, $k \times s_{\ell-1}$, $1 \times s_\ell$, $k \times s_{\ell+1}$, …, $k \times s_n$. If, for the given value of ℓ, these letters are insufficient for any integer $k \ge 0$, print -1 instead.

Examples

standard input	standard output
3 10 NEERC NERC NEF	9 9 -1
4 4 LENSE TEN SENSELESSNESSES LENSE	3 -1 0 3