D. Dense Subsequence

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given a string s, consisting of lowercase English letters, and the integer m.

One should choose some symbols from the given string so that any contiguous subsegment of length m has at least one selected symbol. Note that here we choose positions of symbols, not the symbols themselves.

Then one uses the chosen symbols to form **a new string**. All symbols from the chosen position should be used, but we are allowed to rearrange them in any order.

Formally, we choose a subsequence of indices $1 \le i_1 < i_2 < ... < i_t \le |s|$. The selected sequence must meet the following condition: for every j such that $1 \le j \le |s| - m + 1$, there must be at least one selected index that belongs to the segment $[j, j + m - 1]$, i.e. there should exist a k from 1 to t, such that $j \le i_k \le j + m - 1$.

Then we take any permutation p of the selected indices and form a new string $s_{i_{p_1}} s_{i_{p_2}} ... s_{i_{p_t}}$.

Find the lexicographically smallest string, that can be obtained using this procedure.

Input

The first line of the input contains a single integer m ($1 \le m \le 100\,000$).

The second line contains the string s consisting of lowercase English letters. It is guaranteed that this string is non-empty and its length doesn't exceed $100\,000$. It is also guaranteed that the number m doesn't exceed the length of the string s.

Output

Print the single line containing the lexicographically smallest string, that can be obtained using the procedure described above.

Examples

input
3 cbabc
output
a

input
2 abcab
output
aab

input
3 bcabcbaccba
output
aaabb

Note

In the first sample, one can choose the subsequence $\{3\}$ and form a string "a".

In the second sample, one can choose the subsequence $\{1, 2, 4\}$ (symbols on this positions are 'a', 'b' and 'a') and rearrange the chosen symbols to form a string "aab".