C: Champernowne Subsequence

Time Limit: 2 seconds, Memory limit: 2G

The k^{th} Champernowne word is obtained by writing down the first k positive integers and concatenating them together. For example, the 10^{th} Champernowne word is 12345678910.

It can be proven that, for any finite string of digits, there exists some integer k such that the finite string of digits will appear as a subsequence in the k^{th} Champernowne word.

String s is a subsequence of string t if it is possible to delete some (possibly zero) characters from t to get s.

Given a string of digits, compute the smallest integer k such that the given string of digits is a subsequence of the k^{th} Champernowne word.

Input

The first line of input contains a single integer n ($1 \leq n \leq 10^5$), the length of the string of digits.

The second line of input contains a string of n digits.

Output

Output a single integer k, the minimum integer such that the given string is a subsequence of the k^{th} Champernowne word.

Sample Input 1	Sample Output 1
2 90	10

Sample Input 2	Sample Output 2
2 00	20

This page is intentionally left blank.