

A Menace 2 \$ociety

Input file: **standard input**
Output file: **standard output**
Time limit: 15 seconds
Memory limit: 256 megabytes

Jay and Jay's son ja\$on are playing a game together. This game involves counting occurrences of a string J in another string S . But, ja\$on has gotten so good at this game, Jay is at a loss for how to keep it interesting.

Jay is a menace to society and has decided to make the game more fun for Jay's son ja\$on by adding some new rules. On top of S , Jay will define $K + 1$ more strings S_0, S_1, \dots, S_K . S_i can be created by taking the first character in S , skipping the next i characters, then taking the next character, then skipping the next i characters, and so on. For example, if S is **southpacific**, then S_2 is **staf**. Note that S_0 and S are the same string.

Further, in honour of Jay's son ja\$on's name, Jay will use letters from a language called 'Engli\$h', which contains all lowercase English letters, except that the letter **s** is replaced with the letter **\$**. So **\$outh**, **pacific**, **programming**, **conte\$t** are valid words in Engli\$h, but **south** and **contest** are not.

In Jay's new game, he must find the string J in each of the $K + 1$ words. For example, if S is **aacbc** and J is **ac**, then S_1 is **acc**, and S_2 is **ab**. J is found once in both S_0 and S_1 , but not S_2 .

Given the strings S and J , as well as the value of K , determine the number of occurrences of J in each S_i .

Input

The first line of the input contains a single integer K ($0 \leq K < 1\,000\,000$), which is the number of strings to check.

The second line of the input contains a single non-empty string S that has strictly more than K and at most $1\,000\,000$ characters, consisting of Engli\$h letters.

The third line of the input contains a single non-empty string J , consisting of Engli\$h letters. The length of J is at most the length of S .

Output

Display the number of occurrences of J in S_0, S_1, \dots, S_K .

Examples

standard input	standard output
3 ca\$hca\$h cc	0 0 0 1
0 abc abc	1
2 aacbc ac	1 1 0
3 \$f\$f\$f \$f	3 0 1 0