Problem C
DNA Subsequences

The 3rd Universal Cup, Stage 40: Potyczki. Limits: 1024 MB, 3 s. *21.06.2025*

In this task, we consider sequences of nucleotides in a DNA molecule, which are strings composed of characters '`A`', '`C`', '`G`', and '`T`'. For each natural number k, there are 4^k different k-letter nucleotide sequences. For a fixed natural number k, we say that a given nucleotide sequence s is *k-rich* if all k-letter nucleotide sequences are subsequences of s (not necessarily contiguous).

You are given an n-letter nucleotide sequence s. For each natural number k in the range $[1, n]$, output the minimum number of characters that must be changed in s to make it a k-rich sequence. Note that for each k, the result is calculated independently.

Input

The first line of the input contains an integer n ($1 \le n \le 200\,000$), representing the length of the string s.

The second line of the input contains an n-letter nucleotide sequence s, consisting only of characters '`A`', '`C`', '`G`', and '`T`'.

Output

The output should consist of n integers; the k-th integer should represent the minimum number of characters that must be changed in s to make s a k-rich nucleotide sequence. If it is impossible to change the characters in s in the described way for a given k, then the k-th number should be -1 instead.

Example

For the input data:

```
8
AAGTAGAA
```

the correct result is:

```
1 3 -1 -1 -1 -1 -1 -1
```

Explanation: For $k = 1$, we can change s with one modification to, for example, `AAGTCGAA`. The resulting nucleotide sequence then contains all one-letter words as subsequences (in other words, each of the four letters appears at least once), and thus is 1-rich.

For $k = 2$, we can change s with three modifications to, for example, a 2-rich nucleotide sequence `CAGTTGAC`. Note that we could not change s to, for example, the sequence `CCGTTGAA`, as it does not contain the two-letter word `AC` as a subsequence.

For $k > 2$, it is impossible to make the sequence s k-rich.