

## Problem C

# CheatCode Decoder II

Time limit: 3 seconds

Memory limit: 1024 megabytes

### Problem Description

Morse failed at cheating last time due to some unexpected events and got caught, but after retaking Discrete Mathematics, he finally realized: if there are three distinct items to place into three slots, all possible arrangements are  $3 \times 3 \times 3 = 27$ . Inspired by this, Morse came up with a new method of cheating.

This method takes advantage of the fact that there are 26 lowercase English letters plus a space, exactly 27 symbols, to transmit encrypted messages.

Morse plans to use tapping sounds to send a cipher (暗號). Each cipher symbol (a letter or a space) consists of three sounds, and there are only three types of sounds: A, B, and C. To uniquely encode all 27 symbols using three sounds in three positions, each possible combination is first assigned a code from 0 to 26. The correspondence between a sound combination and a code is determined by the lexicographic order of the combinations.

For example:

- AAA is the first combination in lexicographic order, so it represents 0 (which corresponds to 'a').
- ABB is the fifth combination, so it represents 4 (which corresponds to 'e').
- CCB is the second-to-last combination, so it represents 25 (which corresponds to 'z').
- CCC is the last combination, so it represents 26 (which corresponds to ' ', the space character).

Your task is to help Morse write a decoder. The decoder should allow him to input one or more sequences of three sounds that he hears and determine which symbol each sequence represents – one of the 26 letters (a–z) or the space character ' '.

**Lexicographic order:** This is the order you would find in a dictionary, comparing sequences element by element from left to right. For example, when comparing two three-sound sequences, you first compare the first sound: the one with the earlier letter ( $A < B < C$ ) comes first. If the first sounds are equal, you compare the second sound, and if those are equal too, you finally compare the third sound. This method systematically lists all possible sequences in a predictable, ordered way.

### Input Format

The input contains multiple lines. Each line represents a sequence of sounds of length  $len$  ( $3 \leq len \leq 999$ ). Each sound is one of the three types: A, B or C. Every three consecutive sounds form a single cipher symbol. You may assume that the length of each line is always a multiple of 3.

## Output Format

For each input line, output a single line containing the decoded message. Decode each consecutive group of three sounds into its corresponding symbol (a–z or space).

## Technical Specification

- $3 \leq len \leq 999$
- The length of each line is always a multiple of 3.

### Sample Input 1

```
AAACCCAABAAC
BBABBCBCCCAAABB
ACCBBAACCCACABCCAAACCA
```

### Sample Output 1

```
a bc
morse
im gray
```