Problem 173: Morse Code

Difficulty: Medium

Author: Richard Green, Whiteley, Hampshire, United Kingdom

Originally Published: Code Quest 2022

Problem Background

Morse Code, named for its creator Samuel Morse, is a widely recognized method of encoding text through a series of dots and dashes. This was originally created in the 1800's for use with the telegraph, a device that used electrical pulses to transmit messages, but has also been used with sound, light, and radio signals as well.

When using a telegraph, the operator would push a button to send an electrical signal. The signal would remain active as long as the button was held; releasing the button would end the signal. When transmitting a message, the operator would represent a dot by holding down the button for one unit of time, and a dash by holding down for three units of time. Spaces between dots and dashes within a character would be represented by releasing the button for one unit of time; spaces between characters would be represented by a three-unit delay, and between words a seven-unit delay.

Problem Description

The original Morse Code was designed for efficiency; the amount of time required to transmit each English letter was inversely proportional to the frequency with which it was used. For example, the most common letter in the English language, E, was represented by a single dot. Conversely, Q, one of the least common letters, required two dashes, a dot, and another dash.

With modern computers, efficiency is less of a concern; you and your friends are more concerned about eavesdroppers. You've agreed to create a custom version of Morse Code to use when sending messages back and forth. All you need to do is write a program that can read in the Morse alphabet you'll be using, then encrypt and decrypt messages using that code.

Messages in Morse Code will be represented in text as a series of periods (for dots), dashes, and spaces. Each dot and dash within a letter should be separated by a single space. Letters within a word should be separated by three spaces. Words should be separated by seven spaces.

Sample Input

The first line of your program's input, received from the standard input channel, will contain a positive integer representing the number of test cases. Each test case will include:

- 26 lines representing the Morse Code alphabet to use for this test case, listed in alphabetical order. Each of these lines will begin with an uppercase letter, followed by a space, followed by a combination of periods (.), dashes (-), and spaces, representing the Morse Code representation for that letter.

- A line containing a message written in uppercase letters and spaces that must be encoded using the provided Morse Code.
- A line containing a message written in the provided Morse Code that must be decoded.

```
1
A . -
B - . . .
C - . - .
D - . .
E .
F . . - .
G - - .
H . . . .
I . .
J . - - -
K - . -
L . - . .
M - -
N - .
O - - -
P . - - .
Q - - . -
R . - .
S . . .
T -
U . . -
V . . . -
W . - -
X - . . -
Y - . - -
Z - - . .
HELLO WORLD
. .       . - . .   - - -   . . . -   .       - . - .   - - -   - . .   .       -
- . -   . . -   .   . . .   -
```

(Note that the encrypted message above represents only one line of text; it is too long to fit on this page.)

Sample Output

For each test case, your program must print:

- A line containing the now-encrypted Morse Code message
- A line containing the now-decrypted English message

```
. . . .   .   . - . .   . - . .   - - -       . - -   - - -   . - .   . - . .   -
. .
I LOVE CODE QUEST
```

(Note that the encrypted message above represents only one line of text; it is too long to fit on this page.)