Problem 145: Caesar With a Shift

Difficulty: Medium

Author: Carlos J. Sepulveda, Orlando, Florida, United States

Originally Published: Code Quest 2021

Problem Background

A Caesar Cipher is a simple substitution cipher that replaces each letter in the alphabet with another letter according to a shift value. The most common shift value is 3, so A gets replaced with D, B with E, and so on, wrapping around so that XYZ becomes ABC:

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
DEFGHIJKLMNOPQRSTUVWXYZABC
```

Unfortunately, the cipher is so common and easy to solve that it's not of much use… so let's make things a bit more interesting.

Problem Description

The Caesar Cipher we described above is a monoalphabetic cipher - there's only one replacement for each letter. With our variant, each letter will be encrypted differently based on a set of shift values and directions. Letters can be shifted between 1 and 25 positions, either left (backwards) or right (forwards).

For example, the letter A could be encrypted in several different ways:

- A right shift of 3 produces D, as it would in a traditional Caesar Cipher.
- A left shift of 3 produces X, wrapping around the end of the alphabet.
- A right shift of 10 produces K, the tenth letter in the alphabet after A.
- A left shift of 10 produces Q, ten letters before A (again, wrapping around).

The shifts and directions will be presented as sets of numbers; the shifts between 1 and 25 inclusive, and the directions either 0 (for left) or 1 (for right). The sets may not be the same length, and most likely will be shorter than the message you're trying to decrypt. Whenever you reach the end of a key set, loop around to the beginning of the set again.

Given an encrypted message and the shift and directions sets used to encrypt it, your team will need to reverse the cipher to decrypt the message. Spaces and other punctuation in the message should be left as-is, and do not advance your positions in the key sets.

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 three lines:

- The encrypted message, which contains uppercase letters, spaces, and punctuation, and may be of any length.
- A list of at least two positive integers between 1 and 25 inclusive, separated by spaces, representing the shift values that were used to encrypt the message.
- A list of at least two integers, each either 0 or 1 and separated by spaces, representing the direction values that were used to encrypt the message.

```
3
EDEPCG
1 1 2
1 0 1
EDENEC
1 1 2
1 0
DYA! EZP?
1 2
1 0 0
```

Sample Output

For each test case, your program must print the original, plaintext message. Print the message on a single line in lowercase letters, preserving any spaces or punctuation.

```
decode
decode
cab! car?
```