3. Christina

Program Name: Christina.java **Test Input File: christina.dat**

In a recent computer science class, Christina has learned about string manipulation, and wants to use the names of her friends to practice her newly learned techniques. She decides to list all of her friends in a datafile, input them, and then use a special coding system to combine the names in certain situations to create some unique and interesting name combinations. For example, she'll use the letter 'U' to indicate uppercasing the entire name, 'L' to lowercase it, and 'R' to keep the name in its regular original format. Furthermore, she decides to reverse the order of the letters in the name in a similar fashion by using 'u', 'l', and 'r' which still performs the uppercase, lowercase, or keep regular casing, but reverses the order of the letters in the entire name.

She puts all of the names into a list, and then indicates which name to use with a 01 indicating the first name in the list, and 12 (if there are 12 names) to indicate the last name. In the sample data below, her name is in position 3 of the list of names, and so a code string of "03U" will produce her name in all uppercase letters, resulting in CHRISTINA. She then tries several codes, like "03u12R05L", which produces this result:

ANITSIRHCZacharyeric

The first portion of the code, "03u", causes her name, "Christina", to be shown in all uppercase letters, in reverse letter order. The next part, "12R", takes the 12th name in the list, "Zachary", and keeps it in regular initial case form, normal letter order. Finally, "05L" takes "Eric", and lowercases it, in normal letter order. All codes will consist of three characters, as shown in the examples here.

Input: An initial value N, followed by N names ($1 < N < 100$), all spelled with initial uppercase letters, followed by all lowercased letters. Following all of the names are several code strings which will produce interesting name combinations according to the rules stipulated above.

Output: The resulting name combination string produced by each of the code strings at the end of the data file.

Sample Input:

```
12
Arun
Changpu
Christina
Dai
Eric
Guang
Keerthi
Matthew
Rohan
Sasha
Yulia
Zachary
03u12R05L
01u05L06R
12R07U
10l03R09u
08R09u12l03u11l
12L05U
```

Sample Output:

```
ANITSIRHCZacharyeric
NURAericGuang
ZacharyKEERTHI
ahsasChristinaNAHOR
MatthewNAHORyrahcazANITSIRHCailuy
zacharyERIC
```