

Mohamed and Ali Baba

Ali Baba has a huge amount of gold and diamonds as well as a huge safe. This safe has a special roulette key with some symbols in a circular shape. It can rotate right "Clockwise" or rotate left "Counter-clockwise". Every time Ali Baba rotates the key a number of rotations in any direction a new statement can be read. Start reading the statement from the upper symbol then move clockwise until you reach it again. For example reading your clock symbols will be (1 2 3 4 5 6 7 8 9 10 11). Mohamed (Ali Baba's son) has learnt programming recently. He wants to write a program to generate the final password after all number of rotations. Help Mohamed to solve this easy problem.



Input

Your program will be tested on one or more test cases. The first line of input contains a single integer T ($1 \leq T \leq 30$) indicating the number of test cases. Each test case starts with a string L ($1 \leq |L| \leq 100$). L_i is a symbol in L , where ($'a' \leq L_i \leq 'z'$), ($'A' \leq L_i \leq 'Z'$), ($'0' \leq L_i \leq '9'$). L_i may be also in ($'-', '@', '#', '&'$). The next line contains the number of the key's rotations N ($1 \leq N \leq 50$), followed by N lines each one contains number of moves M ($1 \leq M \leq 2^{57}$) and direction D ('L' for left, 'R' for right).

Output

For each test case, print "Case_#i:_X" where "X" is the final password, "i" is the number of the test case (starting with 1) and "_" is a white space. Each output should be printed in a separate line.

Example

Input:

```
5
#acmASCIS-&-Mohamed-Ali-&-Ali-Baba
4
2L
1R
15L
4R
0123456789
1
2R
ABCDE
```

2
1L
4R
@-#&&
1
125R
01ABcdz#
4
1232R
23L
22R
989R

Output:

Case #1: Mohamed-Ali-&-Ali-Baba#acmASCIS-&-

Case #2: 8901234567

Case #3: CDEAB

Case #4: @-#&&

Case #5: cdz#01AB