Problem B. Button Lock

Time limit: 3 seconds
Memory limit: 512 megabytes

You are standing in front of the room with great treasures. The only thing stopping you is the door with a push-button combination lock. This lock has d buttons with digits from 0 to $d-1$. Whenever you press a button, it stays pushed down. You can not pop back up just one button, but there is a "RESET" button — pressing it pops up all other buttons. Initially, no buttons are pushed down.

The door instantly opens when some specific set of digits is pushed down. Sadly, you don't know the password for it. Having read the documentation for this specific lock, you found out that there are n possible passwords for this particular lock.

Find the shortest sequence of button presses, such that all possible passwords appear at least once during its execution. Any shortest correct sequence of button presses will be accepted.

Input

The first line contains two integers d and n ($1 \le d \le 10$; $1 \le n \le 2^d - 1$). Next n lines describe possible passwords. Each line contains a string s_i of d zeros and ones: for all j from 1 to d the j-th character is `1` iff the button with the digit $j-1$ must be pushed down.

All strings s_i are different, and each string contains at least one `1`.

Output

On the first line, print the number k — the minimum number of button presses. On the second line, print k tokens, describing the sequence. Whenever you press a button with a digit, print that digit. Whenever you press "RESET", print "`R`".

Examples

standard input	standard output
2 2 10 11	2 0 1
3 4 001 111 101 011	6 2 0 R 1 2 0

Note

In the second example, the sequence `1 2 R 2 0 1` is also possible.