

Problem J. Guess Two Strings

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 512 mebibytes

This is an interactive problem.

Jury has chosen two secret binary strings of length N and called them s and t such that s is not lexicographically greater than t . Your task is to find out which strings were chosen. To do that, you can ask the jury to generate up to Q strings. Each such string r will be generated in the following way:

1. start by assigning $r = s$ or $r = t$, choosing one of them randomly with equal probability,
2. randomly select K **distinct** positions in the string r so that each set of K positions has equal probability of being selected,
3. flip the digits at the selected positions in r : change all “0”s to “1”s and all “1”s to “0”s,
4. give the modified string r to you.

Note that s and t don't change during generation of string r .

Your task is to correctly guess s and t .

Interaction Protocol

Initially, you are given a single line with three integers N , K , and Q ($N = 100$, $K = 15$, $Q = 100$): the length of the strings s and t , the number of positions to choose for flipping, and the maximum number of strings which can be generated.

To request another generated string, print a line containing a single “?” to the standard output. After that, you will be given a line containing a string of N binary digits: the newly generated string r . You can make no more than Q such requests.

When you are ready to make a guess, print a line in the format “! s t ” where s and t are two strings of N binary digits each. After that, terminate your program gracefully.

To prevent output buffering, flush the output buffer after each printed line: this can be done by using, for example, `fflush (stdout)` in C or C++, `System.out.flush ()` in Java, `flush (output)` in Pascal, or `sys.stdout.flush ()` in Python. Also, do not forget to terminate each line of output with a newline character.

Example

standard input	standard output
4 1 42	?
1010	?
1110	?
0110	?
0010	?
0000	?
0100	?
0011	?
0111	!
	0010 0110

Example explanation

This example violates the constraints, and is given only to illustrate the process of interaction. All tests in the testing system will satisfy all the constraints from the statement.