Problem H. Recovery

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

Consider an $n \times m$ matrix of ones and zeros. For example, this 4×4:

$$\begin{matrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{matrix}$$

We can compute even parity for each row, and each column. In this case, the row parities are $[0, 1, 1, 0]$ and the column parities are $[1, 0, 0, 1]$ (the parity is 1 if there is an odd number of 1s in the row or column, 0 if the number of 1s is even). Note that the top row is row 1, the bottom row is row n, the leftmost column is column 1, and the rightmost column is column m.

Suppose we lost the original matrix, and only have the row and column parities. Can we recover the original matrix? Unfortunately, we cannot uniquely recover the original matrix, but with some constraints, we can uniquely recover a matrix that fits the bill. Firstly, the recovered matrix must contain as many 1's as possible. Secondly, of all possible recovered matrices with the most 1's, use the one which has the smallest binary value when you start with row 1, concatenate row 2 to the end of row 1, then append row 3, row 4, and so on.

Input

Each input will consist of a single test case. Note that your program may be run multiple times on different inputs. Each test case will consist of exactly two lines. The first line will contain a string R ($1 \le |R| \le 50$), consisting only of the characters 0 and 1. These are the row parities, in order. The second line will contain a string C ($1 \le |C| \le 50$), consisting only of the characters 0 and 1. These are the column parities, in order.

Output

If it is possible to recover the original matrix with the given constraints, then output the matrix as $|R|$ lines of exactly $|C|$ characters, consisting only of 0's and 1's. If it is not possible to recover the original matrix, output -1.

Examples

standard input	standard output
0110 1001	1111 0111 1110 1111
0 1	-1
11 0110	1011 1101