

2 Parity

You are given n binary strings s_1, \dots, s_n , each of the same length m . Along with each s_i you are given a bit b_i . You are also given some nonnegative integer k and want to know whether there exists a subset S of $\{0, 1, \dots, m-1\}$ of size at most k such that for each $i = 1, 2, \dots, n$, the bit b_i is the XOR of the bits of s_i at the indices in S . The s_i are 0-indexed strings. Recall that the XOR of a set of bits is 1 if the number of bits equal to 1 is odd, else the XOR is 0 (in particular, the XOR of an empty set of bits is 0).

For example, if $s_1 = 1010$ and $S = \{0, 3\}$, then b_1 would be 1 (the first bit of s_1) XOR'd with 0 (the last bit of s_1), which is 1.

Given n , k , and the strings s_1, \dots, s_i and their corresponding b_i , find a set S of size at most k which produces the given b_i . You should also detect when no such S exists.

Input

The first line contains n and k , space-separated ($1 \leq n \leq 64$, $0 \leq k \leq 10$). n lines then follow, where the i th line contains s_i , followed by a space, then b_i . In a given test case all strings s_i are of the same length m ($1 \leq m \leq 50$). k will not be bigger than m .

Output

If no set S of size at most k exists producing the given b_i , output -1 followed by a newline. Otherwise, on the first line output the size of a possible S . If the size of that S is not 0, on the second line, output a space-separated list of the indices in S , followed by a newline. If there exist multiple valid S to be output, you can output any one of your choosing.

Example

Sample Input	Sample Output
3 1	1
111 1	1
001 0	
011 1	