

# Problem 2

## A-maze-ing Lakes

*Time limit: 10 second*

An environmental agency is asked to perform a survey of the lakes found in various places. This effort is part of a program to estimate the size and distribution of the lakes depending on the place they're located. At the initial stage of this project, the agency is going to use aerial photographs to perform the estimation. Due to the large number of lakes that will be investigated they decided to use a computer program to help with the task. More specifically, given a 'two color' image that looks like this:



The task is to find the number of the lakes (marked with blue) as well as the surface area of each one of them. The encoded image is given as a 2D array of tiles that can be either a ground tile or a water tile. Each tile in the image corresponds to one square meter of area. Given a single water tile, a lake is defined to cover all the water tiles that are reachable from the initial tile by performing an arbitrary sequence of up/down/left or right moves on water tiles (not diagonally). You can assume that the border tiles found at the edge of the image are always ground tiles.

### Input:

The first line of the input contains two integers  $N$  and  $M$  separated with a single whitespace character ( $1 \leq N \leq 10^3, 1 \leq M \leq 10^3$ ). These define an  $N \times M$  2D array of tiles. The next  $N$  lines of the input are strings of exactly  $M$  characters, which are a combination of `0` to mark a ground tile and `1` to mark a water tile.

### Output:

The first line of the output should be a single integer  $K$  that is the number of distinct lakes found in the data. The second line should have  $K$  space separated integers that are the sizes in square meters of each lake. The integers in the second line (lake sizes) must be in sorted order, from the smallest to the largest.

Sample Input 1	Sample Output 1
6 13 00000000000000 0111101111110 0001101000100 0001101000110 0011000011110 00000000000000	2 10 15
Sample Input 2	Sample Output 2
6 10 00000000000 0111111110 0100000010 0100000010 0111111110 00000000000	1 20