Josip is a strange painter. He wants to paint a picture consisting of N×N pixels, where N is a power of two (1, 2, 4, 8, 16 etc.). Each pixel will be either black or white. Josip already has an idea of how each pixel will be coloured.

This would be no problem if Josip's painting process wasn't strange. He uses the following recursive process:

- If the picture is a single pixel, he colours it the way he intended.

- Otherwise, split the square into four smaller squares and then:
  1. Select one of the four squares and colour it white.
  2. Select one of the three remaining squares and colour it black.
  3. Consider the two remaining squares as new paintings and use the same three-step process on them.

Soon he noticed that it was not possible to convert all his visions to paintings with this process. Your task is to write a program that will paint a picture that differs as little as possible from the desired picture. The difference between two pictures is the number of pairs of pixels in corresponding positions that differ in colour.

INPUT

The first line contains an integer N (1 ≤ N ≤ 512), the size of the picture Josip would like to paint. N will be a power of 2.

Each of the following N lines contains N digits 0 or 1, white and black squares in the target picture.

OUTPUT

On the first line, output the smallest possible difference that can be achieved.

On the next N lines, output a picture that can be painted with Josip's process and achieves the smallest difference. The picture should be in the same format as in the input.

Note: The second part of the output (the picture) may not be unique. Any correct output will be accepted.

SCORING

In test cases worth 50% points, N will be at most 8.

EXAMPLES

input	input	input
4 0001 0001 0011 1110 **output** 1 0001 0001 0011 1111	4 1111 1111 1111 1111 **output** 6 0011 0011 0111 1101	8 01010001 10100011 01010111 10101111 01010111 10100011 01010001 10100000 **output** 16 00000001 00000011 00000111 00001111 11110111 11110011 11110001 11110000