

Problem D. Chessboard

Time limit 1000 ms

Mem limit 262144 kB

Magnus decided to play a classic chess game. Though what he saw in his locker shocked him! His favourite chessboard got broken into 4 pieces, each of size n by n , n is **always odd**. And what's even worse, some squares were of wrong color. j -th square of the i -th row of k -th piece of the board has color $a_{k,i,j}$; 1 being black and 0 being white.

Now Magnus wants to change color of some squares in such a way that he recolors minimum number of squares and obtained pieces form a valid chessboard. Every square has its color different to each of the neighbouring by side squares in a valid board. Its size should be $2n$ by $2n$. You are allowed to move pieces but **not allowed to rotate or flip them**.

Input

The first line contains **odd** integer n ($1 \leq n \leq 100$) — the size of all pieces of the board.

Then 4 segments follow, each describes one piece of the board. Each consists of n lines of n characters; j -th one of i -th line is equal to 1 if the square is black initially and 0 otherwise. Segments are separated by an empty line.

Output

Print one number — minimum number of squares Magnus should recolor to be able to obtain a valid chessboard.

Sample 1

Input	Output
1 0 0 1 0	1

Sample 2

Input	Output
3 101 010 101 101 000 101 010 101 011 010 101 010	2