

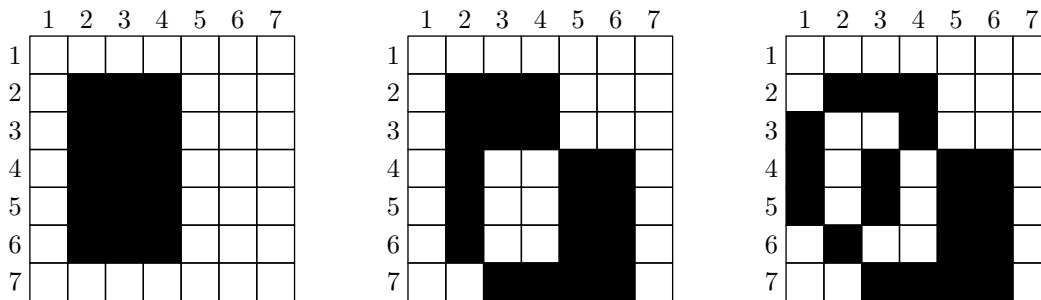
3. Изображения (pilt)

100 очков

Тебя попросили написать программу для мобильного телефона, у которого чёрно-белый экран. Будем считать, что направление увеличения X -координаты этого экрана — слева направо, а направление увеличения Y -координаты — сверху вниз (как это видно на рисунках ниже). Программе для работы нужны разного размера чёрно-белые изображения, которые программа должна отображать на экране, используя графический чип.

В начале работы программы экран целиком белый. Графический чип умеет выполнять только команды типа $XOR(L, R, T, B)$. Выполнение такой команды меняет цвет каждого пикселя в определённом прямоугольнике на противоположный (белые пиксели становятся чёрными, а чёрные — белыми). Верхний левый пиксель этого прямоугольника имеет X -координату L и Y -координату T , а правый нижний пиксель имеет X -координату R и Y -координату B . (Не все графические библиотеки используют для обозначения координат прямоугольника именно такой порядок.)

Рассмотрим для примера получение фигуры на рисунке справа. Применив на полностью белом экране команду $XOR(2, 4, 2, 6)$, получим фигуру как на левом рисунке. Применив далее команду $XOR(3, 6, 4, 7)$, получим фигуру как на рисунке посередине. Наконец, применив команду $XOR(1, 3, 3, 5)$, получим фигуру как на правом рисунке.



Тебе дано несколько чёрно-белых фигур и нужно составить для каждой фигуры набор команд, который, начиная с полностью белого экрана, рисует эту фигуру, используя для этого как можно меньше команд XOR .

Входные данные. Первая строка входных данных содержит число N ($5 \leq N \leq 2000$), которое означает, что размер фигуры составляет $N \times N$ пикселей. На N следующих строках дана фигура, в каждом ряду — ряд пикселей, по порядку сверху вниз. На каждой строке дано N разделённых пробелом целых чисел: цвета пикселей слева направо, где 0 означает белый и 1 — чёрный пиксель.

Выходные данные. В первой строке вывести целое число K ($K \leq 4000$): сколько раз используется команда XOR . На следующих K строках нужно описать команды в порядке их выполнения. На каждой строке вывести параметры одной команды XOR , L_i, R_i, T_i, B_i ($1 \leq L_i \leq R_i \leq N, 1 \leq T_i \leq B_i \leq N$), именно в таком порядке.

Пример.	Входные данные	Выходные данные
	7	3
	0 0 0 0 0 0 0	2 4 2 6
	0 1 1 1 0 0 0	3 6 4 7
	1 0 0 1 0 0 0	1 3 3 5
	1 0 1 0 1 1 0	
	1 0 1 0 1 1 0	
	0 1 0 0 1 1 0	
	0 0 1 1 1 1 0	

Оценивание. В этом задании дано (через систему тестирования) 10 входных файлов, от `input_001.txt` до `input_010.txt`. В качестве решения нужно предоставить соответствующие им файлы вывода, от `output_001.txt` до `output_010.txt`. Программу предоставлять не нужно и она не оценивается.

Некорректное решение (если в файле на самом деле не K команд, или где K или какое-то значение L_i, R_i, T_i, B_i находится за пределами области их допустимых значений, или если данный набор команд не рисует нужную фигуру) получает 0 очков.

Корректное решение каждого теста приносит $1 + 9 \cdot K_B / K_Y$ очков, где K_B — это самый лучший результат на этом тесте среди всех участников соревнования, а K_Y — количество команд, используемое в твоём решении.

Обрати внимание, что во время соревнования система тестирования даёт информацию только о корректности решения, на самом деле сравнение решений участников и оценивание производится в конце соревнования.