

## Задача Е. Черно-белый квадрат

|                          |              |
|--------------------------|--------------|
| Имя входного файла:      | bwsquare.in  |
| Имя выходного файла:     | bwsquare.out |
| Ограничение по времени:  | 1 секунда    |
| Ограничение по памяти:   | 256 мегабайт |
| Отображение результатов: | полное       |

Известный художник Квазимир Малевич нарисовал картину под названием «Чёрно-белый квадрат». Картина имеет вид клетчатого прямоугольника, некоторые клетки которого покрашены в белый цвет, а некоторые — в черный.

Искусствовед Эрик Полоскин выдвинул гипотезу, что художник рисовал картину следующим образом: взял изначально белый холст, после чего проводил на нем вертикальные и горизонтальные полосы черного цвета от края до края холста, толщиной в одну клетку.

Ассоциация Вычислительного Искусствоведения хочет выяснить, может ли эта гипотеза быть верной. Если она верна, то также необходимо найти минимальное количество полос, которые должен был провести Квазимир, чтобы написать данную картину. Кроме того нужно придумать способ рисования данной картины, использующий данное число полос.

### Формат входных данных

Первая строка входного файла содержит два целых числа  $h$  и  $w$  ( $1 \leq h, w \leq 50$ ) — высоту и ширину картины в клетках.

Каждая из следующих  $h$  строк содержит  $w$  чисел через пробел, каждое число — либо 0, либо 1, что означает белую и черную клетку соответственно.

### Формат выходных данных

Если гипотеза Полоскина не может быть верна, выведите единственное число -1.

Если гипотеза может быть верна, выведите на первой строке число  $n$  — минимальное количество полос, которое должен был провести Квазимир, чтобы нарисовать картину.

Во второй строке выведите  $n$  чисел, которые описывают способ рисования данной картины с использованием  $n$  полос. Положительное число  $x$  означает проведение черной вертикальной полосы в  $x$ -м столбце (столбцы нумеруются слева направо, начиная с 1). Отрицательное число  $-x$  означает проведение черной горизонтальной полосы в  $x$ -й строке (строки нумеруются сверху вниз, начиная с 1).

Если существуют несколько способов нарисовать картину с использованием  $n$  полос, выведите любой из них.

### Замечание

Тесты с ответом -1 (с номерами с 23 по 27) будут оценены только в случае прохождения хотя бы одного теста из группы тестов с номерами с 12 по 22.

### Примеры

| bwsquare.in                          | bwsquare.out   |
|--------------------------------------|----------------|
| 3 4<br>1 0 1 0<br>1 1 1 1<br>1 1 1 1 | 4<br>1 3 -2 -3 |
| 2 2<br>1 0<br>0 0                    | -1             |

## Задача F. Королевская прогулка

|                          |              |
|--------------------------|--------------|
| Имя входного файла:      | kingwalk.in  |
| Имя выходного файла:     | kingwalk.out |
| Ограничение по времени:  | 1 секунда    |
| Ограничение по памяти:   | 256 мегабайт |
| Отображение результатов: | нет          |

Шахматный король хочет совершить прогулку по Весеннему полю. Весеннее поле представляет собой прямоугольник, каждая клетка которого содержит одну букву латинского алфавита.

Король начнет прогулку в произвольной клетке поля и сделает  $n - 1$  ходов; каждый ход происходит по правилам хода шахматного короля (в клетку, соседнюю с текущей по стороне или по вершине). Нельзя вместо хода оставаться на той же клетке. Во время прогулки разрешается посещать одну и ту же клетку более одного раза.

Таким образом, король посетит  $n$  клеток. Если рассмотреть буквы из этих клеток в порядке их обхода, то получится строка  $s$ , состоящая из  $n$  букв.

Вечером король сравнит эту строку с девизом его династии, в котором тоже ровно  $n$  букв. Если в  $i$ -й позиции в строке  $s$  и в  $i$ -й позиции в девизе стоит одна и та же буква, то король повысит жалование  $i$ -й пешке.

Помогите пешкам подсказать королю такой маршрут, при котором как можно больше пешек получат повышение.

### Формат входных данных

Первая строка входного файла содержит два целых числа  $h$  и  $w$  ( $2 \leq h, w \leq 20$ ) — размеры Весеннего поля.

Далее следуют  $h$  строк, каждая из которых содержит  $w$  строчных латинских букв без пробелов, — это описание Весеннего поля.

Затем следует строка, содержащая целое число  $n$  ( $1 \leq n \leq 50$ ).

В последней строке содержится девиз королевской династии — строка из  $n$  строчных букв латинского алфавита.

### Формат выходных данных

В первой строке выведите  $m$  — максимальное возможное число пешек, получающих повышение.

В следующих  $n$  строках выведите координаты клеток в том порядке, в котором их должен обходить король, чтобы  $m$  пешек получили повышение.

Координаты клетки — это номер строки (строки нумеруются сверху вниз, начиная с 1) и номер столбца (столбцы нумеруются слева направо, начиная с 1).

Если оптимальных маршрутов несколько, выведите любой из них.

### Примеры

| kingwalk.in   | kingwalk.out                                |
|---|---|
| 3 9<br>zhautykov<br>olympiadk<br>azakhstan<br>6<br>almaty | 4<br>3 1<br>2 2<br>1 3<br>1 4<br>1 5<br>1 6 |
| 3 9<br>zhautykov<br>olympiadk<br>azakhstan<br>6<br>astana | 6<br>2 7<br>3 6<br>3 7<br>3 8<br>3 9<br>3 8 |

## Задача G. Прыжки в высоту

|                          |                         |
|--------------------------|-------------------------|
| Имя входного файла:      | <code>jumper.in</code>  |
| Имя выходного файла:     | <code>jumper.out</code> |
| Ограничение по времени:  | 1 секунда               |
| Ограничение по памяти:   | 256 мегабайт            |
| Отображение результатов: | частичное               |

$N$  спортсменов, пронумерованных от 1 до  $N$ , участвуют в соревнованиях по прыжкам в высоту. Правила соревнований следующие.

Спортсмену даётся по три попытки на каждой высоте. Попытка считается удачной, если в результате прыжка планка удержалась на стойках. Каждая попытка проходит для всех участвующих в ней спортсменов следующим образом: спортсмены, которые ещё не взяли эту высоту и не выбыли из соревнований, прыгают в порядке возрастания номеров. Только в случае трёх неудачных попыток на очередной высоте спортсмен заканчивает соревнование. Переносить попытки на следующую высоту, а также пропускать взятие очередной высоты или очередную попытку не допускается.

Победителем считается тот, кто взял максимальную высоту. Если два и более спортсмена взяли одну и ту же высоту, то они разделяются:

1. по количеству попыток при взятии своей максимальной высоты;
2. по количеству попыток при взятии предыдущей, а при необходимости и более ранних высот;
3. по своим номерам (меньший номер соответствует участнику с лучшим «рейтингом»).

В секторе для прыжков было установлено электронное устройство, фиксирующее номера спортсменов. В результате известен порядок, в котором они совершали свои прыжки. Напишите программу, которая определит вероятную тройку победителей соревнований.

### Формат входных данных

В первой строке входного файла записаны два числа  $N$  и  $M$ , где  $N$  ( $3 \leq N \leq 1000$ ) — количество спортсменов, а  $M$  ( $9 \leq M \leq 100\,000$ ) — количество прыжков, сделанных всеми спортсменами во время соревнования. В следующей строке записаны  $M$  чисел — номера участников в порядке выполнения ими прыжков (как удачных, так и неудачных).

### Формат выходных данных

Выведите в выходной файл три числа — номера участников, которые могли занять первое, второе и третье места соответственно.

### Примеры

| <code>jumper.in</code>                      | <code>jumper.out</code> |
|---|-------------------------|
| 5 18<br>1 2 3 4 5 2 3 4 5 2 3 5 1 4 1 4 1 4 | 1 4 2                   |

### Пояснение к примеру

В примере первый и четвертый спортсмены взяли начальную высоту, а остальные — нет. Но первый взял высоту с более ранней попытки, поэтому и стал победителем. Третье место занял спортсмен с минимальным из оставшихся номеров — 2.

## Задача Н. Болельщики

Имя входного файла: fans.in  
Имя выходного файла: fans.out  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт  
Отображение результатов: частичное

В городе  $M$  футбольные болельщики разделились на фанатов двух команд  $MC$  и  $MF$ . Частные жилые дома в этом городе построены вдоль одной улицы. Мэрии надоело наблюдать за столкновениями фанатов, поэтому был издан указ о переселении болельщиков разных команд в разные концы улицы. Так, все  $K$  болельщиков команды  $MC$  вместе со своими семьями согласно указу должны занимать дома с номерами от 1 до  $K$ , а все  $M$  болельщиков команды  $MF$  вместе со своими семьями должны занимать дома с номерами от  $N - M + 1$  до  $N$ , где  $N$  — общее количество домов и, соответственно, семей в городе. Остальные семьи должны поселиться в домах с номерами от  $K + 1$  до  $N - M$ .

Сотрудникам мэрии для выполнения указа было велено подготовить план переселения, удовлетворяющий следующим условиям. У каждого дома в городе есть своя стоимость. Чтобы компенсировать неудобство семье, переезжающей в более дешевый дом, ей будет выплачена полная стоимость оставляемого дома, а переезжающей в более дорогой дом или дом той же стоимости — не выплачивается ничего. Кроме того, каждая семья переезжает не более одного раза.

Напишите программу, которая определяет минимально возможную сумму денег, необходимую для организации переселения.

### Формат входных данных

В первой строке входного файла задано число  $N$  ( $2 \leq N \leq 300$ ) — количество домов (семей в городе). Далее следуют  $N$  строк, в каждой из которых через пробел записаны два числа целых числа — стоимость соответствующего дома и число 0, 1 или 2, обозначающее принадлежность хотя бы одного жителя этого дома к фанатам одной из двух команд. Так, нулём обозначается отсутствие фанатов в доме, единицей — присутствие фанатов команды  $MC$ , двойкой — присутствие фанатов команды  $MF$ . Гарантируется, что фанаты обеих команд в городе есть. Стоимость домов — положительное целое число, не превосходящее 1000.

### Формат выходных данных

Выведите в выходной файл минимальную сумму денег, необходимую для всех переселений.

### Примеры

| fans.in                                     | fans.out |
|---|----------|
| 3<br>1 2<br>2 1<br>3 0                      | 5        |
| 6<br>5 0<br>3 1<br>6 2<br>5 1<br>2 1<br>1 2 | 6        |

### Пояснения к примерам

В первом примере семья из третьего дома вынуждена переехать во второй, за что ей будет выплачена компенсация в размере трех единиц. Вторая семья переезжает в первый дом и получает компенсацию 2, первая семья бесплатно переезжает в третий дом.

Во втором примере жители первого и четвертого домов обмениваются между собой без компенсаций. Семья из третьего дома обменивается с семьёй из пятого дома и получает компенсацию 6.