

## RAMSINSP: City Inspection

Легенда, описанная далее, переформулирована и упрощена переводчиком, чтобы читатель мог лучше понять условие задачи. Оригинальную легенду вы можете прочитать на странице задачи в контексте.

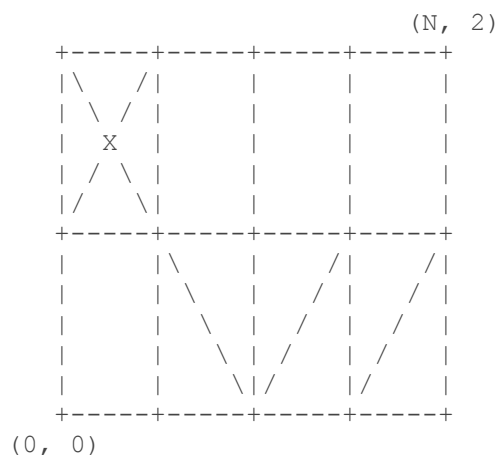
### Условие:

Chef знаменит своими великолепными рецептами. Он хочет открыть новый ресторан в городе ManiLand.

Город ManiLand имеет очень строгую структуру в виде решетки. Город имеет три главных дороги, ведущих с запада на восток, и  $N+1$  более коротких второстепенных дорог, ведущих с юга на север. Первая главная дорога идет от перекрестка  $(0, 0)$  до  $(N, 0)$ , вторая – от  $(0, 1)$  до  $(N, 1)$ , третья – от  $(0, 2)$  до  $(N, 2)$ .  $i$ -тая второстепенная дорога ( $0 \leq i \leq N$ ) идет от перекрестка  $(i, 0)$  до  $(i, 2)$ . Кроме того, существуют магистральные дороги, которые проложены диагонально от какого-либо перекрестка  $(x, y)$  до  $(x+1, y+1)$ , или от какого-либо  $(x, y)$  до  $(x+1, y-1)$  ( $x$  и  $y$  – целые числа). Все дороги – двусторонние.

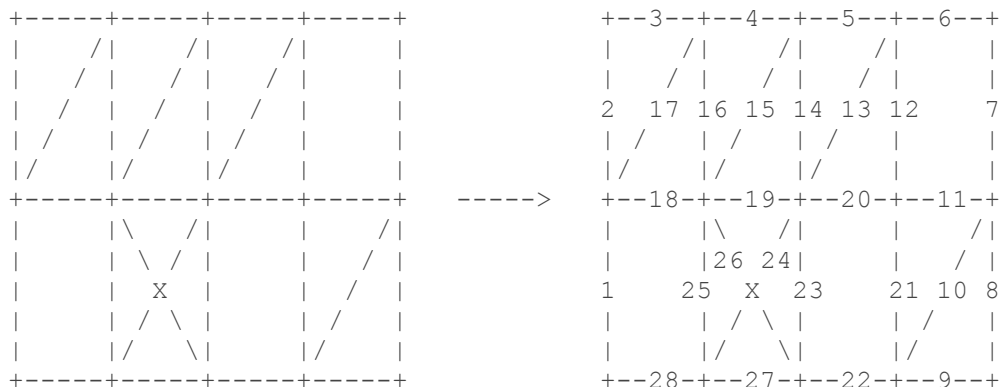
Определим *квартал* как квадратную область, заданную 4 углами:  $(x, y)$ ,  $(x, y+1)$ ,  $(x+1, y+1)$ ,  $(x+1, y)$  для любых целых  $x$  и  $y$ . ManiLand имеет  $N$  кварталов в длину и  $2$  квартала в ширину.

На следующем рисунке изображен пример планировки города (обратите внимание, что магистрали в одном квартале могут пересекаться).



Полицейский участок расположен на перекрестке  $(0, 0)$ . Каждый день полицейские должны патрулировать дороги. Чтобы сделать это, необходимо проехать по каждому участку дороги хотя бы один раз. Тем не менее, полицейские заметили, что часто при окончании патрулирования получается, что они проезжали некоторые дороги (или их части) более одного раза.

Конкретная планировка города называется *хорошо патрулируемой*, если возможно выехать из полицейского участка, проехать по всем частям всех дорог ровно  $1$  раз (за исключением, возможно, пересечений) и вернуться обратно в полицейский участок. На рисунке выше изображена планировка не хорошо патрулируемого города. На следующем рисунке изображена планировка хорошо патрулируемого города:



(Каждая дорога проезжается ровно один раз. Некоторые пересечения посещаются дважды, но это не имеет значения.)

Теперь правительство желает сделать ManiLand хорошо патрулируемым. Для этого планируется разрушить несколько магистральных дорог и построить новые, а главные и второстепенные дороги оставить нетронутыми.

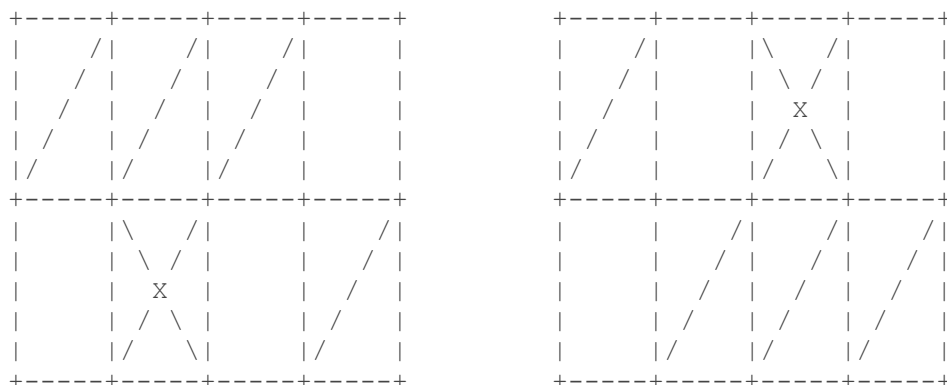
Стоимость разрушения и строительства магистрали, идущей от  $(x, y)$  до  $(x+1, y+1)$ , равна  $H_b$  и  $H_d$  соответственно, и стоимость разрушения и строительства магистрали, идущей от  $(x, y)$  до  $(x+1, y-1)$ , равна  $L_b$  и  $L_d$  соответственно. Правительство хочет узнать минимальную необходимую стоимость для того, чтобы сделать город ManiLand хорошо патрулируемым, и количество способов это сделать с минимальной стоимостью.

Магистрали должны удовлетворять следующим требованиям:

- Все магистрали должны идти диагонально от какого-либо перекрестка  $(x, y)$  до  $(x+1, y+1)$  или от какого-либо  $(x, y)$  до  $(x+1, y-1)$ ;
- Магистрали не могут вести за пределы решетки.
- Никакие две магистрали не должны соединять одинаковую пару перекрестков.

Порядок разрушения и строительства магистралей не имеет значения. Два способа считаются различными, если планировка города в результате будет различной.

Для первого примера положим  $H_b = 20$ ,  $H_d = 15$ ,  $L_b = 4$ ,  $L_d = 31$ . Тогда минимальная стоимость перестройки равна **106** и существует **2** способа сделать планировку хорошо патрулируемой за эту стоимость:



Правительство наняло Chef'a для решения этой задачи. Но так как Chef занят открытием своего нового ресторана, он просит Вас помочь решить эту задачу за него.

Отметим, что ManiLand может быть очень длинным (т.к.  $N$  – очень большое число), но, к счастью, планировка города – предсказуема. Вся планировка города длины  $N$  состоит из частей

длины  $K$ , которые повторяются  $N/K$  раз. (Обратите внимание, что вы не должны поддерживать эту повторяющуюся структуру при попытке сделать город хорошо патрулируемым).

### **Формат ввода:**

Первая строка содержит целое число  $T$  – количество тестов.

Далее следует описание тестов в следующем формате:

Первая строка каждого теста содержит разделенные пробелом целые числа  $H_b$ ,  $H_d$ ,  $L_b$ ,  $L_d$ ,  $N$  и  $K$  – стоимость разрушения и строительства магистралей как описано выше, длина города и длина повторяющегося участка.

Вторая и третья строка содержат строку длины  $K$ , состоящую из символов  $'/'$ ,  $'\backslash'$ ,  $'X'$  и  $'.'$  – описание повторяющегося участка.  $'/'$  и  $'\backslash'$  представляют квартал с одной магистралью,  $'X'$  представляет собой квартал с двумя магистралями, и  $'.'$  представляет собой блок без каких-либо магистралей. Смотрите пример ввода для получения дополнительной информации.

### **Формат вывода:**

Для каждого тестового случая выведите в отдельную строку два разделенных пробелом целых числа – минимальную необходимую стоимость и количество способов сделать город хорошо патрулируемым за минимальную стоимость. Так как количество способов может быть очень большим, выведите его по модулю  $10^9+7$ .

Если это невозможно, выведите в отдельную строку **“a kitchen nightmare”** (без кавычек).

### **Ограничения & оценивание:**

- $1 \leq T \leq 10^5$
- $1 \leq K \leq 10^5$
- $1 \leq N \leq 10^{15}$
- Сумма всех  $K$  в одном входном файле не превышает  $10^5$
- $K$  делит  $N$
- $1 \leq H_b, H_d, L_b, L_d \leq 1000$

### **Примеры тестов:**

#### **Входные данные:**

```
3
20 15 4 31 4 4
X...
.\//
1 3 4 10 2 1
X
/
11 11 11 11 1 1
.
.
```

#### **Выходные данные:**

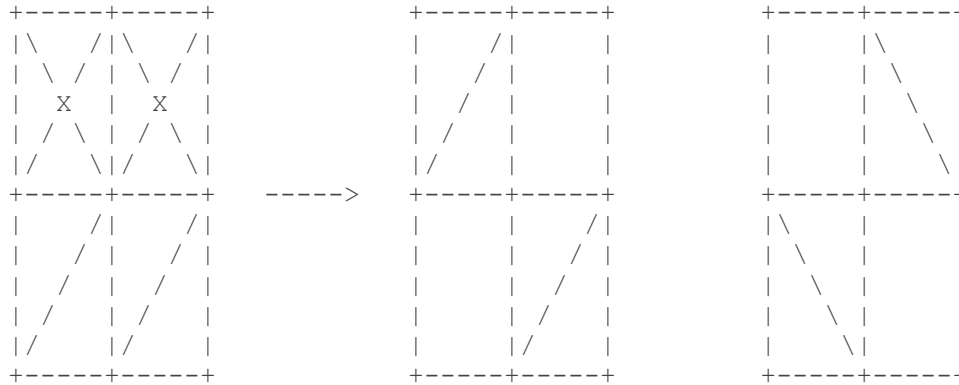
```
106 2
26 2
a kitchen nightmare
```

### **Пояснения:**

1

**Тест 1:** Пример описан в условии.

**Тест 2:**  $N = 2$ ,  $K = 1$ , следовательно,  $N/K = 2$  – данный шаблон повторится два раза. Существует два способа сделать данную планировку хорошо патрулируемой с минимальной стоимостью **26**:



**Тест 3:** При любом варианте строительства магистралей невозможно сделать город хорошо патрулируемым.