

## Задача 6. Морской бой

Имя входного файла:	<i>стандартный ввод</i>
Имя выходного файла:	<i>стандартный вывод</i>
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 МБ

В рамках Чемпионата Урала планируется проведение турнира стратегий по игре «Морской бой 1D».

Игра проходит на поле, которое представляет собой прямоугольник размером  $1 \times N$  клеток. На поле расставляются  $T$  кораблей, каждый из которых имеет вид прямоугольника размером  $1 \times K$  клеток. Расстановка кораблей на поле является *допустимой*, если различные корабли не имеют общих клеток и разделены хотя бы одной пустой клеткой. Игровая программа осуществляет выстрелы в клетки поля, а сервер сообщает, является ли выстрел промахом или попаданием в корабль.

В процессе игры про некоторые клетки становится известно, что при любой допустимой расстановке кораблей они принадлежат какому-либо из кораблей. Назовём такие клетки *заведомо занятыми*.

Игра заканчивается после первого попадания в корабль. Сервер пытается добиться того, чтобы игра продолжалась как можно дольше. Для этого он не фиксирует расстановку кораблей в начале игры, а рассматривает все возможные допустимые расстановки и сообщает о попадании, только если клетка, в которую осуществляется выстрел, является заведомо занятой.

Требуется написать программу, исполняющую роль сервера для этой игры. Сервер сначала загружает параметры игры, а затем взаимодействует с игровой программой, сообщая после каждого выстрела информацию о промахе или попадании, а также количество заведомо занятых клеток.

### Протокол взаимодействия

**Задача является интерактивной.** После каждого вывода требуется сбросить буфер вывода.

Роль игровой программы исполняет программа жюри. Программа-решение исполняет роль сервера.

Первая строка стандартного ввода программы-решения содержит параметры игры — три числа:  $N$  — размер игрового поля,  $T$  — число кораблей и  $K$  — длина каждого корабля ( $1 \leq N \leq 100\,000$ ,  $1 \leq T$ ,  $1 \leq K$ ). Гарантируется, что на поле длины  $N$  можно по описанным правилам разместить  $T$  кораблей длины  $K$ .

После считывания параметров игры программа-решение должна определить и вывести в стандартный поток вывода количество заведомо занятых клеток.

Затем начинается игра. Программа-решение должна последовательно считывать ходы игровой программы из стандартного потока ввода и обрабатывать их следующим образом:

1. Считать из стандартного потока ввода одно число  $q$  — номер клетки, в которую игровая программа осуществляет выстрел ( $1 \leq q \leq N$ ). Игровая программа никогда не делает два выстрела в одну и ту же клетку.
2. Если клетка  $q$  является заведомо занятой, вывести в стандартный поток вывода число 1 и завершить работу.
3. Если клетка  $q$  не является заведомо занятой, вывести в стандартный поток два числа, разделенных пробелом: 0 и количество заведомо занятых клеток после этого выстрела.

После этого программа-решение переходит к пункту 1 и продолжает взаимодействие с игровой программой.

### Система оценивания

Данная задача содержит три подзадачи. Для оценки каждой подзадачи используется своя группа тестов. Баллы за подзадачу начисляются только в том случае, если все тесты из этой группы успешно пройдены.

#### Подзадача 1

$N \leq 15$ . Подзадача оценивается в 30 баллов.

#### Подзадача 2

$N \leq 3000$ . Подзадача оценивается в 30 баллов.

#### Подзадача 3

$N \leq 100\,000$ . Подзадача оценивается в 40 баллов.

## Пример взаимодействия

стандартный ввод	стандартный вывод
8 2 3	4
4	0 5
1	1

## Пояснение к примеру

Игра происходит на поле из 8 клеток, на котором расставляются 2 корабля, состоящие из 3-х клеток каждый. Все допустимые расстановки кораблей приведены на рис. 1. Клетки, отмеченные «#», заведомо заняты. Таких клеток 4.

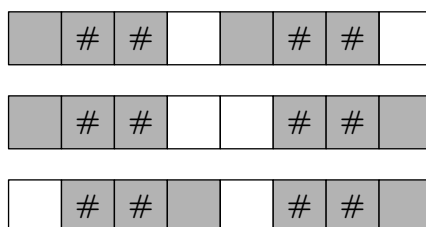


Рис. 1. Допустимые расстановки кораблей в начале игры.

Первый выстрел производится в клетку с номером 4. Это выстрел считается промахом, остаются допустимыми расстановки кораблей, приведенные на рис. 2. Теперь 5 клеток заведомо заняты.

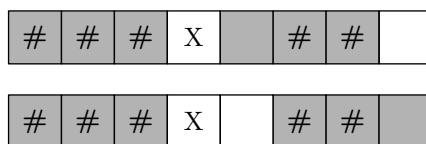


Рис 2. Допустимые расстановки кораблей после первого выстрела.

Второй выстрел производится в клетку с номером 1, эта клетка не может быть заведомо занятой, поэтому игра завершается.