

Programación de Aviones

La mayoría de los aeropuertos tienen un gran terreno llano para que aterricen y despeguen los aviones. Aquí tienes un ejemplo, en el que el terreno llano está dividido en 4 filas y 5 columnas. Los cuadrados grises son lugares para aterrizar y despegar, los negros son obstáculos y los cuadrados con números son plazas de aparcamiento.

01				02
03	04		05	06
07	08			09
10	11			12

A cada avión se le asigna una plaza de aparcamiento. Puede elegir una casilla gris arbitraria para aterrizar, y luego se dirige a su plaza de aparcamiento asignada mediante una secuencia de movimientos horizontales y verticales (cada movimiento consiste en desplazarse una casilla hacia el norte, el sur, el este o el oeste). No puede moverse en diagonal, ni puede moverse a una casilla obstáculo o a una plaza de aparcamiento que esté ocupada por otro avión. Las casillas vacías sin números son siempre libres para moverse. Después de llegar a su plaza de aparcamiento asignada, el avión espera hasta que llega el momento de despegar. Entonces, se dirige a una casilla gris arbitraria para despegar (no necesariamente aquella en la que aterrizó). Inicialmente, el terreno llano está vacío. Una asignación es factible si se pueden realizar todos los aterrizajes y despegues. Obsérvese que se pueden asignar distintos aviones a la misma plaza de aparcamiento, siempre que el horario sea factible.

La lista de eventos se representa mediante una secuencia de números enteros, donde positivo significa aterrizaje y negativo despegue. Se garantiza que cada avión aterriza y despegue exactamente una vez (al final, el terreno plano vuelve a estar vacío).

Por ejemplo, la lista de sucesos +1, +2, +3, +4, +5, +6, -6, -5, -4, -3, -2, -1 tiene una asignación factible 12, 09, 05, 06, 02, 10, que son las plazas de aparcamiento asignadas al avión 1, 2, 3, 4, 5, 6, respectivamente. Escribe un programa para asignar plazas de aparcamiento a los aviones.

Entrada

La entrada consiste en un máximo de 20 casos de prueba. Cada caso comienza con una línea que contiene n , r y c

($0 < n < 21, 2 < r, c < 11$) donde n es el número de aviones, r y c son el número de filas y columnas del terreno llano. Cada una de las siguientes r líneas contiene c pares de caracteres separados por un espacio. Cada par es un espacio de aterrizaje (==), un espacio vacío(..), un cuadrado de obstáculo(##) o una plaza de aparcamiento(dos dígitos). Cada plaza de aparcamiento tiene un número diferente. El último caso va seguido de un único cero, que no debe procesarse.

Salida

Para cada caso de prueba, imprima el número de caso y si hay una solución. Si la hay, la segunda línea debe contener n números enteros de dos dígitos (con ceros a la izquierda si los hay), el número de plaza de aparcamiento asignada del avión correspondiente. Si existe más de una solución, cualquiera es aceptable.

Ejemplo de entrada

6 4 5

01 .. == .. 02 03 04 .. 05 06

07 08 .. ## 09

10 11 ... 12

+1 +2 +3 +4 +5 +6 -6 -5 -4 -3 -2 -1

3 3 3

== .. 01

.. ## 02

.. ## 03

+1 +2 +3 -3 -1 -2

0

Ejemplo de Salida

Case 1: Yes

12 09 05 06 02 10

Case 2: No