

## Recuperando el tesoro

*Contribución de Jorge Arroyuelo*

### Descripción del problema

Hay que recuperar un tesoro que fue oculto en una cueva. La cueva consta de  $M \times N$  ( $1 \leq N$ ,  $M \leq 120$ ) habitaciones cuadradas, cada una conectada mediante una puerta con sus adyacentes (con las que comparte una pared). Como las habitaciones forman una grilla, podemos distinguir cada una de ellas mediante sus coordenadas de la forma:  $(m,n)$ ; donde  $m$  y  $n$  indican el número de fila y columna respectivamente ( $0 \leq m \leq M-1$ ,  $0 \leq n \leq N-1$ ). Puede verse en la figura el orden de numeración de las habitaciones.

En las habitaciones puede haber pozos o monstruos que viven allí. El objetivo es encontrar el camino que se debe recorrer a través de las habitaciones, sin caer en una que contenga un pozo o un monstruo, hasta llegar al tesoro. Ya que es posible la existencia de varios caminos hasta el tesoro, necesitamos encontrar aquel que sea el más corto posible.

La persona encargada de ir en busca del tesoro podrá llevar una cantidad  $F$  ( $0 \leq F \leq 120$ ) de flechas, las cuales poseen una sustancia anestésica. Haciendo uso de estas flechas se podrá pasar por aquellas habitaciones donde se encuentre un monstruo, sin dañarlo, ya que simplemente se lo duerme por varias horas, otorgando tiempo suficiente para salir de la cueva una vez recuperado el tesoro antes de que despierte. Para pasar por una habitación donde se encuentre un monstruo se debe gastar exactamente una flecha. Por las habitaciones donde se encuentra un pozo es imposible pasar. Sólo puede ingresar al conjunto de habitaciones por la  $(0,0)$ .

Por lo tanto, se te pide que escribas un programa `tesoro.cpp`, `tesoro.c` o `tesoro.pas` que dada la cantidad de flechas que podrán utilizarse como máximo, el mapa de la cueva indicando donde se encuentra el tesoro y las habitaciones donde están los pozos y monstruos, determines el camino más corto que puede seguirse para llegar al tesoro y poder recuperarlo. Para encontrar el camino requerido puedes utilizar tantas flechas como creas conveniente, sin

sobrepasar las recibidas. No se te pide que ahorres flechas.

### Datos de entrada

Se recibe un archivo `tesoro.in` con el siguiente formato:

- Una línea con 3 números:  $M$ ,  $N$  y  $F$ ; que indican las dimensiones de la cueva y la cantidad de flechas disponibles para la misión.
- $M$  líneas con  $N$  letras yuxtapuestas, donde la letra 'W' significa que existe un monstruo en la habitación correspondiente, la letra 'P' indica la existencia de un pozo en esa habitación, la letra 'V' que no hay ni pozo ni monstruo y la letra 'T' donde se encuentra el tesoro. La primera línea que se lee es la 0 y así siguiendo.

### Datos de salida

Se debe generar un archivo `tesoro.out` conteniendo:

- Una línea indicando la cantidad de habitaciones por las que se debe pasar.
- Una línea por cada habitación del recorrido, indicando las coordenadas (de la forma  $(m, n)$  indicada anteriormente) correspondientes a cada una de ellas. El recorrido comienza siempre en la habitación  $(0, 0)$ , por lo que no es necesario incluirla en el archivo de salida ni contarla como parte del recorrido.

En caso de no existir un camino hasta el tesoro, de las características descriptas, se debe indicar en el archivo de salida una única línea con el texto: "imposible".

### Puntuación

El puntaje se reparte 50/50 entre la primera línea (cantidad de celdas visitadas) y las siguientes (enumeración de las celdas visitadas).

( continúa en otra hoja )

**Ejemplo**

Si el mapa fuera el de la figura, y además se cuenta con una única flecha, la entrada `tesoro.in` sería:

```
6 12 1
VVPVVVVPVVV
VPVWVVPVPTW
VWVVVPVWVVV
VVVWVVVPVWV
VVWVVVVVVVV
VPVPVWVWVVP
```

La salida `tesoro.out` podría ser:

```
18
(0, 0)
(1, 0)
(2, 0)
(3, 0)
(3, 1)
(3, 2)
(3, 3)
(4, 3)
(4, 4)
(4, 5)
(4, 6)
(4, 7)
(3, 7)
(2, 7)
(2, 8)
(2, 9)
(2, 10)
(1, 10)
```



	0	1	2	3	4	5	6	7	8	9	10	11
0												
1												
2												
3												
4												
5												