

CAMPO MINADO II

(**Nota preliminar:** este enunciado es similar al problema CAMPO MINADO I, sólo que en vez de comprobar si un camino dado es solución válida, ahora se pide generar una solución.)

Se nos pide que dirijamos a un robot de un cierto punto a otro en un recinto rectangular delimitado por cuatro paredes, cada una de ellas orientada a uno de los puntos cardinales (en sentido horario: N, E, S y O).

El robot está diseñado para desplazarse utilizando una cuadrícula que cubre el suelo del recinto, de modo que puede avanzar en cada paso desde una casilla hasta cualquiera de las que tenga un lado en común con ella. Así, podemos considerar el espacio por el que puede moverse el robot dividido en m filas (numeradas de 1 a m , yendo de N a S) y n columnas (numeradas de 1 a n , yendo de O a E).

Para gobernar el desplazamiento del robot sólo se dispone de tres órdenes: "giro a la izquierda" (I), "giro a la derecha" (D) y "avance"(A). El avance se realiza de casilla en casilla (una cada vez), pudiéndose sólo pasar - como ya hemos dicho - desde la casilla en la que se encuentre el robot a cada una de las cuatro (o menos) que tienen un lado en común con esta. En todo momento el robot está orientado directamente hacia uno de los cuatro lados de la casilla en la que se encuentra (que distinguiremos nombrándolos como los puntos cardinales; es decir, en sentido horario: N, E, S y O), y las órdenes I y D hacen girar al robot 90° sin avanzar. De este modo, si por ejemplo el robot tiene orientación N, con la orden I, quedaría en la misma casilla pero con orientación O (pues el giro hacia la izquierda corresponde al giro antihorario); y con la orden D quedaría orientado al E (pues el giro a la derecha es el que se realiza en sentido horario). La orden de avance se aplica siempre en la orientación que tenga en ese momento el robot y ha de tenerse en cuenta que un intento de avanzar contra los límites del recinto provocaría la destrucción del robot.

El problema es que el espacio por el que ha de desplazarse el robot se encuentra minado, de forma que si el robot pisa alguna de las casillas que tiene una mina será destruido y no podrá completar su misión. Lo mismo sucede si el robot se estrella contra alguna de las paredes del recinto al recibir una orden de avance incorrecta, como ya hemos dicho.

Objetivo

Se trata de dirigir un robot de un cierto punto a otro dentro del recinto sin que sea destruido (siempre que sea posible), usando para ello una secuencia adecuada de órdenes.

Entrada

La entrada del programa consiste de una secuencia de líneas, que residen en un archivo de texto (ASCII) con nombre CAMPO2.IN, que tendrá el siguiente formato:

- La primera línea (la 1) contiene el número de filas (m) y columnas (n) del recinto separados por una coma y sin espacios en blanco. Podéis suponer que se cumple $1 \leq m < 100$ y $1 \leq n < 100$.
- Las m líneas siguientes (de la 2 a la $m+1$) contienen, cada una, una fila del recinto. Concretamente, la línea tendrá n ceros o unos separados por comas y sin espacios en blanco entre ellos. Un 0 corresponde a una casilla que puede ser pisada sin peligro y un 1 corresponde a una casilla minada.
- La siguiente línea (la $m+2$) contiene las coordenadas (i_1, j_1), $1 \leq i_1 \leq m$, $1 \leq j_1 \leq n$, de la posición inicial del robot, separadas con una coma y sin espacios en blanco.
- La siguiente línea (la $m+3$) contiene las coordenadas (i_2, j_2), $1 \leq i_2 \leq m$, $1 \leq j_2 \leq n$, del destino del robot, separadas con una coma y sin espacios en blanco.
- La última línea (la $m+4$) contiene una única letra indicando la orientación inicial del robot (N, E, S o O).

Salida

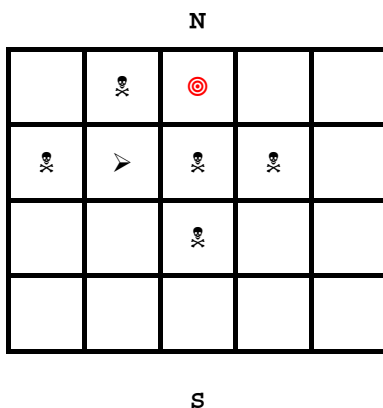
La salida del programa ha de grabarse en un archivo de texto (ASCII) con nombre CAMPO2.OUT, que contendrá la secuencia de órdenes para el robot, escritas una por línea. Cada línea del archivo contendrá por tanto una única letra mayúscula, a saber: "D", "I" o "A". Todas las líneas deberán estar correctamente finalizadas con saltos de línea y no podrán contener ningún otro carácter.

Si, por la situación de las minas, fuera imposible completar la misión, se deberá grabar en este archivo una línea con el texto "MISION IMPOSIBLE" (tal cual está escrito, en mayúsculas, sin acentos y sin espacios adicionales). Ha de notarse que en general existirán muchas posibles soluciones. El programa puede dar cualquiera de ellas (sólo una). Sólo en caso de empate entre concursantes, se tendría en cuenta la longitud de la misma (contada como número de órdenes tipo "A" que hay que dar al robot). Lo que sí debería evitarse es generar soluciones que hagan pasar al robot dos veces por el mismo sitio (pues esto es siempre innecesario), y también giros innecesariamente largos (por ejemplo, pasar de "N" a "E" mediante tres giros "I" en vez de un único giro "D").

EJEMPLO:

CAMPO2.IN	CAMPO2.OUT
4,5	D
0,1,0,0,0	A
1,0,1,1,0	A
0,0,1,0,0	I
0,0,0,0,0	A
2,2	A
1,3	I
E	A
	D
	A
	I
	A
	A
	I
	A
	A

El fichero de entrada correspondería al siguiente recinto
(circulo = destino; flecha = robot orientado; calaveras = posiciones minadas):



En CAMPO2.OUT se muestra una posible solución (existen otras, una de ellas igual de "corta").