

ANÁLISIS Y DISEÑO DE ALGORITMOS

Práctica 7 de laboratorio

El problema del laberinto

Se dispone de una cuadrícula $n \times m$ de valores $\{0, 1\}$ que representa un laberinto. Un valor 0 en una casilla cualquiera de la cuadrícula indica una posición inaccesible; por el contrario, con el valor 1 se simbolizan las casillas accesibles. Por ejemplo:

$\xrightarrow{(1,1)}$	1	1	0	0	1
	1	1	1	1	1
	0	1	1	0	0
	1	1	0	1	1
	1	1	1	0	0
	0	0	0	1	1 $\xrightarrow{(6,5)}$

Se pide, aplicar el **método voraz** para tratar de encontrar el camino de longitud mínima¹ que conduzca a la salida del laberinto. El punto de partida es la casilla $(1, 1)$ y el de llegada la casilla (n, m) . Sólo son validos tres tipos de movimientos desde una casilla cualquiera (i, j) :

1. derecha: $(i, j + 1)$,
2. abajo: $(i + 1, j)$,
3. abajo y derecha (diagonal): $(i + 1, j + 1)$.

Como es evidente, tampoco son válidos los movimientos que llevan al exterior del laberinto ni los que conducen a casillas inaccesibles. Por otra parte, de entre los movimientos posibles se debe escoger aquel que siga un criterio voraz, es decir, aquel que mejor contribuya a minimizar la longitud del camino resultante².

Salida del programa:

Deberá mostrarse por pantalla la etiqueta **Camino:** seguida de la secuencia de casillas que se van visitando. En la siguiente línea se mostrará la etiqueta **Longitud:** seguida de la longitud del camino encontrado hasta la salida. En el caso de que el criterio voraz mencionado no encuentre camino hasta la salida deberá mostrarse la etiqueta **No_encuentro_salida** al final de la secuencia de casillas visitadas. En este caso como valor longitud no se debe mostrar nada.

Ejemplos de posibles salidas a dos instancias distintas del problema:

- Camino: (1,1) (2,1) (3,2) (4,3)
Longitud: 4
- (1,1) (2,2) (3,3) (3,4) No_encuentro_salida
Longitud:

El primer ejemplo sería el caso en el que se encuentra un camino de salida, es decir, se alcanza la casilla destino. El segundo ejemplo sería el caso en el que no es posible continuar desde la última casilla visitada, por lo que no se ha encontrado ningún camino de salida.

¹Se entiende por longitud del camino el número de casillas que lo componen.

²Debe tenerse en cuenta que el método voraz no garantiza, para este problema, encontrar el camino mínimo ni siquiera que se encuentre algún camino aunque éste exista.

Entrada al programa:

Al igual que en la práctica anterior, el laberinto se suministrará codificado en un fichero de texto cuyo nombre se recogerá a través de la línea de comando, por ejemplo: **laberinto entrada.txt** donde entrada.txt es el nombre del fichero que contiene el laberinto a resolver. Su formato y contenido será:

- Línea 1 del fichero: valores n y m separados mediante un único espacio en blanco.
- Línea 2 (y siguientes): m valores $\{0,1\}$ que componen la primera fila (y siguientes) del laberinto, separados mediante un único espacio en blanco

por tanto, el fichero contendrá $n + 1$ líneas, todas ellas (incluso la última) finalizarán con un salto de línea.

Ejemplo 1:

Sea el siguiente contenido de un fichero de texto. Se trata de un laberinto (6×6):

6	6				
1	1	0	0	1	0
1	1	1	1	1	0
0	1	1	1	0	0
1	1	1	0	1	0
1	1	1	1	1	0
0	0	1	1	1	1

La salida podría ser:

Camino: (1,1) (2,2) (3,3) (3,4) (4,5) (5,5) (6,6)
Longitud: 7

Ejemplo 2:

Laberinto:

6	5				
1	1	0	0	1	
1	1	1	1	1	
0	1	1	0	0	
1	1	0	1	1	
1	1	1	1	0	
0	0	0	1	1	

La salida podría ser³:

Camino: (1,1) (2,2) 3,3) (4,4) (4,5) No_encuentro_salida
Longitud:

³En este caso, si se hubiera priorizado el movimiento hacia abajo en lugar de hacia la derecha sí se hubiera encontrado un camino de longitud 6 que también seguiría un criterio voraz. En general, priorizar uno u otro movimiento de entre estos dos resulta equivalente.