



Análisis y Diseño de Algoritmos

Grado en Ingeniería Informática

Práctica 2 - curso 2023/24

Introducción

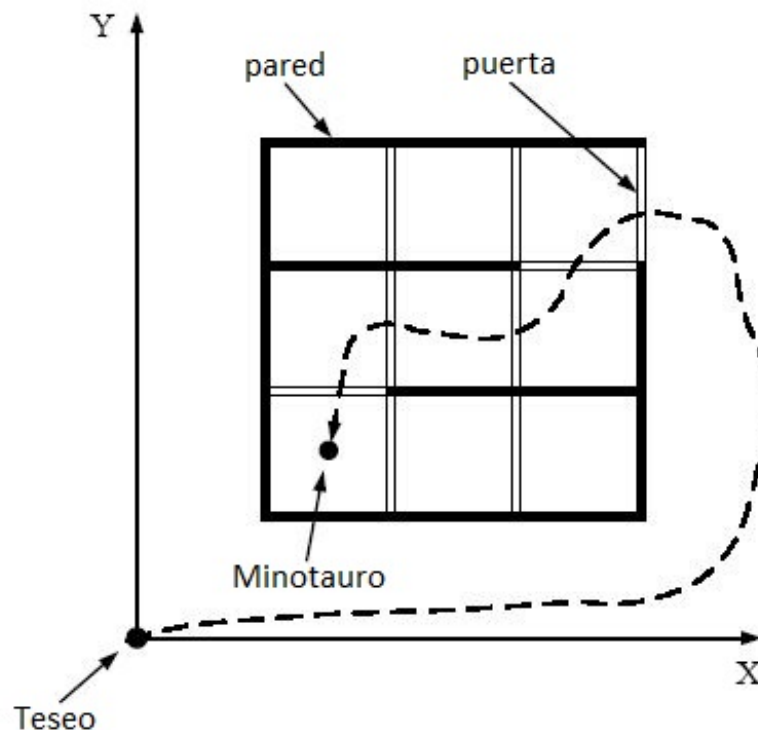
En la isla de Creta hay un laberinto en el que está encerrado el Minotauro. La misión de Teseo es entrar en el laberinto y localizar al Minotauro para matarlo.

Tras consultar el mapa, Teseo descubre que el laberinto tiene paredes y puertas. Todas las paredes son paralelas al eje X o al eje Y. Se supone que el grosor de las paredes es cero. Todas las puertas se abren en las paredes y tienen una longitud de 1. Teseo no puede atravesar una pared a menos que haya una puerta en ella. Como atravesar una puerta es peligroso, Teseo quiere atravesar el menor número posible de puertas para encontrar al Minotauro.

Suponemos que la posición inicial de Teseo está en $(0, 0)$. Dada la posición del Minotauro y la configuración de paredes y puertas, el objetivo de la práctica es **escribir un programa que calcule el número mínimo de puertas que tiene que atravesar Teseo para llegar al Minotauro**.

En la evaluación de la práctica **se valorará la eficiencia en tiempo de ejecución del algoritmo desarrollado**.

En la siguiente figura se muestra un ejemplo del laberinto y el camino que recorrió Teseo para encontrar al Minotauro:





Formato de entrada y salida

Se deberá seguir estrictamente el formato de entrada y salida descrito a continuación, según se muestra en el ejemplo de ejecución.

El programa leerá los datos de entrada del fichero de texto `entrada.txt`. El archivo de entrada consta de varios casos de prueba. La primera línea de un caso de prueba contiene dos números enteros no negativos M y N . M representa el número de paredes del laberinto y N el número de puertas. A continuación siguen M líneas, cada una de las cuales contiene cuatro enteros que describen una pared en el siguiente formato:

$x \ y \ d \ t$

(x, y) indica el punto inferior izquierdo de la pared, d es la dirección de la pared: “0” significa que es paralela al eje X y “1” que es paralela al eje Y. Por último, t da la longitud de la pared. Las coordenadas de los dos extremos de cualquier muro estarán en el intervalo $[1, 199]$.

Luego hay N líneas que dan la descripción de las puertas:

$x \ y \ d$

x, y, d tienen el mismo significado que para las paredes. Como las puertas tienen una longitud fija de 1, se omite t .

La última línea de cada caso contiene dos números reales positivos:

$f1 \ f2$

$(f1, f2)$ da la posición del Minotauro. Y no se encontrará dentro de ninguna pared o puerta.

Un caso de prueba de $M = -1$ y $N = -1$ indica el final de la entrada, y no debe procesarse.

El programa deberá imprimir por pantalla la solución encontrada. Para cada caso de prueba, en una línea separada, hay que indicar el número mínimo de puertas que Teseo debe atravesar para asesinar al Minotauro. Si no puede alcanzar al Minotauro, hay que indicar ‘-1’.



Ejemplo de ejecución

- Entrada (fichero `entrada.txt`):

```
8 9
1 1 1 3
2 1 1 3
3 1 1 3
4 1 1 3
1 1 0 3
1 2 0 3
1 3 0 3
1 4 0 3
2 1 1
2 2 1
2 3 1
3 1 1
3 2 1
3 3 1
1 2 0
3 3 0
4 3 1
1.5 1.5
4 0
1 1 0 1
1 1 1 1
2 1 1 1
1 2 0 1
1.5 1.7
-1 -1
```

- Salida:

```
5
-1
```

Normas de entrega

La práctica se realizará en **grupos de 2 personas**. Se deberá entregar el código fuente de los programas que se han creado para la realización de la práctica (se recomienda el uso del lenguaje Java aunque alternativamente se permite realizar en cualquier lenguaje de programación). Dichos programas deben compilar y ejecutar perfectamente en las máquinas de los laboratorios. Además se entregará un documento en **formato PDF** donde se indique lo siguiente:

- Nombre de los alumnos.



Universidad de Valladolid

Departamento de Informática

Escuela de Ingeniería Informática

Campus Miguel Delibes, s/n. 47011 Valladolid

Tel.: 983423670 Fax: 983423671

- **Muy importante:** hay que estimar como se ha repartido el trabajo entre los miembros del equipo. Para ello, **hay que indicar el porcentaje de esta entrega que se estima ha desarrollado cada miembro del equipo.**
- Una breve descripción de cómo se ha realizado la práctica.
- Análisis de la eficiencia del programa desarrollado.

La fecha límite de entrega es el **15 de diciembre a las 14:00**. Los ficheros de la práctica se empaquetarán en un fichero zip cuyo nombre será `p2_grupoX.zip`, siendo X el número del grupo de laboratorio. El fichero zip se entregará a través del Campus Virtual (<https://campusvirtual.uva.es/>).

La defensa de la práctica es obligatoria y se realizará en la sesión de laboratorio de la semana siguiente a la entrega:

- Grupo L1: miércoles 20 de diciembre a las 10:00.
- Grupo L2: jueves 21 de diciembre a las 12:00.