**Ingeniería del conocimiento**

***Memoria Práctica 1***

El objetivo de esta memoria es detallar las particularidades de la práctica implementada de forma clara y concisa.

* **Detalles generales**

Solo se ha implementado la **funcionalidad básica** de la práctica. Es posible, dado un tablero determinado, indicar una casilla de inicio, una casilla de destino, y varios bloques los cuales no podrán ser traspasados en la ruta final que lleve desde el inicio hasta el destino. El correcto funcionamiento de la práctica sólo se garantiza si se indica exactamente una posición de inicio, una posición de destino y ninguno o varios bloques intraspasables. En caso de que estas premisas no se cumplan, no se garantiza el correcto funcionamiento de la práctica.

La práctica **carece de interfaz gráfico**, la interacción se resume a **ficheros de texto**, tanto de entrada como de salida. El fichero de entrada tiene que estar situado en la misma ruta en la que se encuentra el ejecutable de la práctica y debe llamarse exactamente “board.txt” para que sea detectado por el ejecutable. El fichero de salida que muestra el camino solución se generará en un fichero llamado “solucion.txt” que se generará también en la misma ruta en la que se encuentre el ejecutable.

El tablero que se reciba como fichero de entrada, tiene que tener que ser cuadrado, es decir, de longitud **NxN** pudiendo tomar N como máximo el valor 50. Este valor es el que tiene por defecto la práctica, aunque puede ser fácilmente ampliable si es preciso.

* **Detalles de implementación**

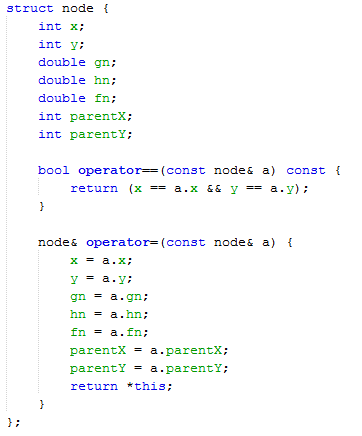
El lenguaje elegido para la implementación de la práctica es **C++**.

El tablero se representa internamente como un array bidimensional de enteros. Se ha creado un fichero de cabecera explícitamente para el tablero. Este fichero contiene la **clase tablero**, con su **constructora por defecto** y **dos operaciones**, una que pone un elemento concreto en una posición determinada dada una fila y una columna y el elemento que se quiere poner en esa posición , y otra que obtiene el elemento situado en una posición concreta dada una fila y una columna. Internamente la traducción de enteros a elementos en las distintas posiciones del tablero es la siguiente:

|  |  |
| --- | --- |
| **Valor entero** | **Significado** |
| -1 | Muro, casilla que no puede ser atravesada |
| 1 | Casilla normal, que si puede ser atravesada |
| 2 | Casilla inicial |
| 3 | Casilla final |

El fichero de texto que representa el tablero de entrada está compuesto mayoritariamente por **caracteres** que se traducirán internamente a los números enteros anteriormente descritos.

Se ha creado una estructura específica para la aplicación del algoritmo A estrella. La estructura es la siguiente:



La estructura *nodo* almacena las coordenadas de la casilla del tablero (la **columna** es **x**, y la **fila** es **y**), el coste actual hasta la posición (**hn**), la distancia euclidea hasta la casilla destino (**gn**), y la suma de ambas (**fn**), así como las coordenadas del nodo padre (**parentX** y **parentY**), es decir, del nodo anterior al actual. Se han implementado las operaciones de asignación y comparación para dotar de mayor flexibilidad a la estructura *nodo* y así facilitar su uso.

Gracias a esta implementación con *nodos* es posible obtener el camino solución de forma muy sencilla. Una vez se cierra el *nodo* de fin, simplemente hay que ir recorriendo la estructura *nodo*, accediendo al padre de cada *nodo* (repitiendo está operación cuantas veces sea necesaria) hasta llegar al *nodo* en el que empieza el camino, es decir, el *nodo* de inicio.