Memoria Práctica 1 IC

# Detalles de implementación

Lenguaje utilizado **Python.** El procedimiento llevado a cabo ha sido el desarrollo de un algoritmo estrella siguiendo el procedimiento explicado en clase, con una interfaz gráfica desarrollada mediante pygame, un conjunto de módulos de Python para el desarrollo de videojuegos. Hemos generado una matriz para recorrer que es un array bidimensional de elementos de clase celda que contienen su posición y la de sus vecinos posteriormente.

**Explicación del algoritmo:** El algoritmo funciona mediante el uso de la función h que será nuestra función que estima la distancia desde el nodo en el que estamos al nodo destino calculando la diferencia entre sus respectivas posiciones “y” y “x” en valor absoluto. En la función correspondiente al A\* esta reibe la función para dibujar, la matriz, el origen y el destino. Generamos una priority queue para la lista abierta y en ella incluimos el valor de la función f, el orden de entrada de esa celda(el orden de entrada es para desempatar entre dos caminos con la misma f) y la celda en sí. Después tendremos un mapa clave valor para guardar las g’s y otro para las f’s. Además tendremos un mapa clave valor que nos marque la celda anterior a la que estemos para poder reconstruir el camino. Hecho esto sólo sería partir del nodo origen e ir accediendo a los vecinos introduciéndolos en la lista abierta y sacando los que tengan el menor valor de f al ser una priority queue la lista abierta (recordemos la f como la suma de g + h, es decir el recorrido actual más el que estimamos que queda). Cuando accedemos a un nodo calculamos su g y su f y los guardamos en los mapas y en la lista open. Así hasta que recorriendo de manera estimada las zonas que pueden ser recorridos óptimos hasta llegar al final. Una vez llegamos al final con el mapa que nos marque la celda anterior vamos dibujando el recorrido óptimo de final a comienzo.

El punto extra añadido han sido los Way Points, las metas intermedias antes de llegar al destino.

# Manual de usuario

El programa consta de 6 botones principales:

**Empezar:** Al pulsar este botón se ejecutará el programa con la configuración actual del tablero, se dibujará el recorrido más corto de la casilla de origen a la casilla destino.

**Salir:** Permite cerrar el programa

**Origin:** Permite situar el punto de origen donde se desee dentro del tablero, al hacer click sobre este botón después haciendo click en cualquier lugar del tablero se situará la casilla de origen.

**Objective:** Su funcionamiento es el mismo que el de el botón origin pero con la casilla de destino.

**Obstacle:**  Al clickar en el botón de obstacle se podrá colocar un obstáculo clickando en cualquier casilla del tablero, estos obstáculos pueden ser varios y no se pueden retirar, para ello tendrá que usar el botón reset.

**Way Point:** Al clickar en este botón se puede situar un punto intermedio al que se deba llegar antes de llegar al objetivo, según se sitúen se seguirá en orden el recorrido hacia estos objetivos intermedios.

**Reset:** Reinicia el tablero situando el origen y el destino en posiciones aleatorias y eliminando los obstáculos existentes, al igual que el trazado de recorrido que se haya podido dibujar por ejecutar el programa.