

Prácticas Pathfinding

Práctica 1

Utilizad el motor de prácticas y el proyecto de ejemplo para implementar un pathfinding sobre un grid.

Para gestionar el grid podéis simplemente pintar cuadrados de debug para representar las casillas libres, las ocupadas, las de mayor coste, etc.

Dentro del proyecto de ejemplo tenéis una clase Pathfinder definida en pathfinder.h/cpp donde tenéis definidas las funciones que os servirán como puntos de entrada al pathfinder.

Las funciones SetStartPosition y SetEndPosition sirven para poner las posiciones de inicio del path y de objetivo del path respectivamente. Se llaman tanto al principio del programa desde test.lua (pathfinder:setStartPosition(5, 10), pathfinder:setEndPosition(20, 40)), como con eventos de ratón. El botón izquierdo llama a SetStartPosition con las coordenadas donde estuviera el ratón y el botón derecho llama a SetEndPosition.

Estas funciones ponen las variables miembro m_StartPosition y m_EndPosition a sus valores (que pueden obtenerse con GetStartPosition y GetEndPosition). Y también se llama a la función UpdatePath.

La función miembro UpdatePath tiene que calcular un path desde GetStartPosition, hasta GetEndPosition.

El resultado de la práctica es:

- Calcular path en UpdatePath
- Pintar path en DebugDraw
- Pintar grid del escenario:
 - En DebugDraw

Los datos de entrada son:

- Fichero con el mapa que define el grid lógico (los datos que necesita el pathfinding)

Opciones para el fichero del mapa del grid lógico:

- Fichero de texto con un carácter para cada casilla: '.' si es pasable, # si no es pasable.
- Fichero de texto con un carácter para cada casilla, con una letra que representa el coste de la casilla
 - A: coste 1
 - B: coste 2
 - C: coste 3
 - D: coste 4
 - ...
- Fichero de texto con un carácter para cada casilla como el anterior, pero con un fichero adicional definiendo el coste de cada tipo de casilla. Esta es la mejor opción ya que ofrece la

mayor flexibilidad posible para configurar el pathfinding sólo con datos. Sólo con modificar los ficheros de texto podríamos implementar cualquier juego basado en grids.

Opcionales

Práctica 2

Cuando tenemos que depurar un proceso de pathfinding es super útil poder ejecutarlo paso a paso para ver cómo va abriendo nuevos nodos y va cerrando los antiguos.

En esta práctica vamos a hacer que el pathfinding se haga paso a paso. Cada pulsación de la barra espaciadora llamará a la función PathfindStep.

Esta función realizará una iteración del bucle del algoritmo normal de A*, si termina el proceso construye el path y devuelve true. Con lo que la función UpdatePath lo que hace es poner el estado inicial y llamar repetidamente a PathfindStep hasta que devuelva true.

La función DrawDebug pintará el path y también los nodos abiertos y cerrados.