Robert Gentile

EGP-410

Assignment4

The current Pathfinding project was built upon the Dean Lawson’s original Pathfinding structure. For modifying the project, the intent was to maintain the pre-existing architecture as much as possible to implement the new path finding methods. This intent especially so regarding the lower level graphic methods like the visualizer, and most classes in the CommonpathfindingLib solution. The most notable change in the CommonpathfindingLib is that in the grid class the directions for adjacent tiles to exclude diagonal movement.

For the new Pathfinding methods, they were made to be inherited from gridPathfinder like that of the pre-existing Depth-first method. Like the Depth-first method the Find path for A\* and Dijkstra follow the same structure of passing through nodes. The key difference being that connection costs are taken into account for visited node placement in the new methods. To find the best path determined by the Pathfinders Nodes were changed to store their previous node id so that when the goal was reached the drawing functions could back trace starting from the goal towards the start. Originally I had made a vector of paths that branched with each round of found connections. However, I scrapped this due to the inefficiency of created so many paths that are majority unused. To accommodate the back tracing the Path class was given new getter functions to access nodes and their indices.

Finally, the GameApp class was given new functions to switch its pathfinding member and the classes that access it. The functions are similar to the changing behaviors in steering, where a new pointer is created and passed into a setter function. And of course the Inputsystem calls a new message to call these functions in game.