**README**

I have implemented Breadth First Search (BFS), Depth First Search (DFS), and A\* Search. I have also made two versions of each. One which expands in a preset order regardless of the direction to the goal, and another which tries to expand toward the goal before trying other directions.

The script “Project\_2.m” allows the user to select the search method, including the expansion order, and the weighting for A\*, if desired. The script then generates random start and goal node positions, runs the selected search function, and saves the path node positions and a picture of the display.

If you want to use the BFS algorithm, then set the variable “METHOD” to [1,0,0].

If you want to use the DFS algorithm, then set the variable “METHOD” to [0,1,0].

If you want to use the A\* algorithm, then set the variable “METHOD” to [0,0,1].

If you want to use the direction to the goal to control the order in which nodes are expanded, then set the variable “smart” to 1. If not, then set to 0.

To specify the weight for [weighted] A\* change the variable “weight” to a value other than 1. When A\* is not the selected method, it does not matter what weight is set to.

For example, if one used the following settings: “METHOD = [1,0,0]; smart = 1; weight = 1;” then the function called would be: “Path = Breadth\_First\_Search\_2 (StartNode, GoalNode, METHOD)” which uses BFS and tries to expand toward the goal first.

Note: “METHOD” is required for the specific search algorithm functions because I did not want to have too many algorithm specific sub-functions, but some algorithm specific sub-functions called within non-specific sub-functions, and must be specified.