**Brief Overview**

This path planning program uses an algorithm called Improved Q-Learning (IQL). A-Star uses a combination of heuristic value (h-value) along with distance traversed (g-value) to come up with a f-value that helps determine the path for a robot to take. IQL instead assigns each node a Q-value with the goal having the largest Q-value. Using the Q-values of each node as a guide, the robot is able to find a path from start to goal. IQL is an improvement over the Classical Q-Learning (CQL) algorithm. In CQL, all nodes are given an initial Q-value of 0. This means that the robot must randomly choose a node to go to until it finds the goal. However, this IQL attempts to get over this flaw by using an algorithm called flower pollination. This algorithm is used to mimic actual pollination in nature. Nodes/Flowers are chosen at random and pollen is distributed amongst them using local and global pollintaion equations for a set number of iterations. This way before the actual Q-learing starts, some nodes are already initialized with a Q-value meaning that the robot does not have to choose randomly between nodes to find the goal.

**Important Points Regarding the Program**

The original algorithm assumed the robot could only go in the four cardinal directions, however, in this program the robot can move diagonally at a 45 degree angle as well. The enivroment takes a while to create the larger the enviornment is. For the 300x300 enviornment, it may take up to a minute for the enivornment to be created. The larger the enviroment, the harder it is to make out the individual nodes, ensure that you view the graph at maximum size so that you may better see the nodes. It is possible for a path not to be found especially if there are more obstacles or if the starting and/or goal index are situated in the corner of the enviornment.

**Hypothesis**

My hypothesis regarding the results is that the algorithm would take longer the larger the enviornment was as well as take longer if the start and goal points were further apart.

**Results**

Given below are the results of various program runs. The time taken includes the time taken to run the algorithm as well as the time taken to plot the path.



















**Interpretation of Results**

The results for the most part shows that the time taken increases the greater the path distance. The results also seem to show that the obstacle percentage begins to have more of an impact on time taken the greather the path distance is.

**Screenshots**

Here are some of the indices that were tested along with their respective graphs. The blue node represents the starting point and the green node represents the goal. The red color color is used to represent the nodes and path taken by the robot to go from start to finish.







































































