Gokul Srinivasan 500594698

MOTION PLANNING
ASSIGNMENT – 02

Advanced Algorithm Implementation

- **Explain in your own words, how does D* replan a path by updating the cost?**

  D* maintains a list of nodes to be evaluated, known as the "OPEN list". Nodes are marked as having one of several states:
  - NEW, meaning it has never been placed on the OPEN list
  - OPEN, meaning it is currently on the OPEN list
  - CLOSED, meaning it is no longer on the OPEN list
  - RAISE, indicating its cost is higher than the last time it was on the OPEN list
  - LOWER, indicating its cost is lower than the last time it was on the OPEN list

  The algorithm works by iteratively selecting a node from the OPEN list and evaluating it. It then propagates the node's changes to all of the neighboring nodes and places them on the OPEN list. D* begins by searching backward from the goal node.

  Each expanded node has a back pointer which refers to the next node leading to the target, and each node knows the exact cost to the target. When the start node is the next node to be expanded, the algorithm is done, and the path to the goal can be found by simply following the back pointers.

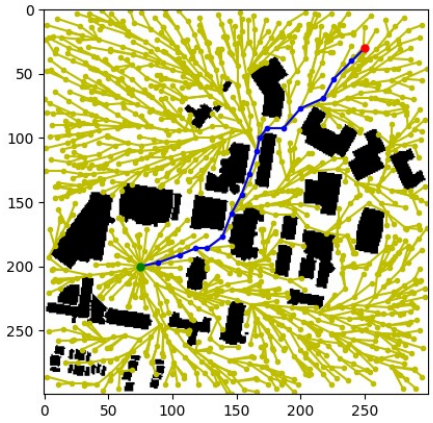
  When an obstruction is detected along the intended path, all the points that are affected are again placed on the OPEN list, this time marked RAISE. Before a RAISED node increases in cost, the algorithm checks its neighbors and examines whether it can reduce the node's cost. If not, the RAISE state is propagated to all of the nodesthat have back pointers to it. These nodes are then evaluated, and the RAISE state is passed on. When a RAISED node can be reduced, its back pointer is updated and passes the LOWER state to its neighbors.

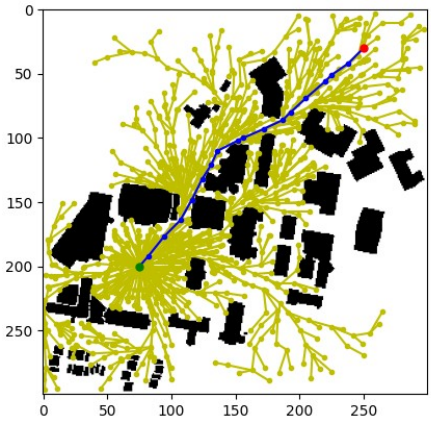
Gokul Srinivasan 500594698

- **Why does D* can replan faster than A* or Dijkstra?**
  A* and Dijkstra are used for static environments, while D* is used for dynamic environments. D* is incremental. When the initial route gets blocked, an incremental search algorithm takes advantage of the previous calculations. In the case of A* and Dijkstra, the environment has to be reevaluated and the path has to be recomputed from scratch. Thus making A* and Dijkstra computationally more expensive than D*. Therefore, D* is comparatively faster than A* and the Dijkstra algorithm.

- **What is the key differences between regular RRT* and informed RRT*?By showing and comparing the results of RRT* and informed RRT*, what is the advantages of using the latter?**

RRT*	Informed RRT*
New point is randomly sampled from the entire workspace	New point randomly sampled from the ellipsoid region.
More nodes have to be explored due to the size of workspace	Less sampling region results in less number of nodes to explore.
The path length is not used to restrict the sampling process.	The path length as the major axis of ellipsoid and is used to restrict the sampling process.





Gokul Srinivasan 500594698

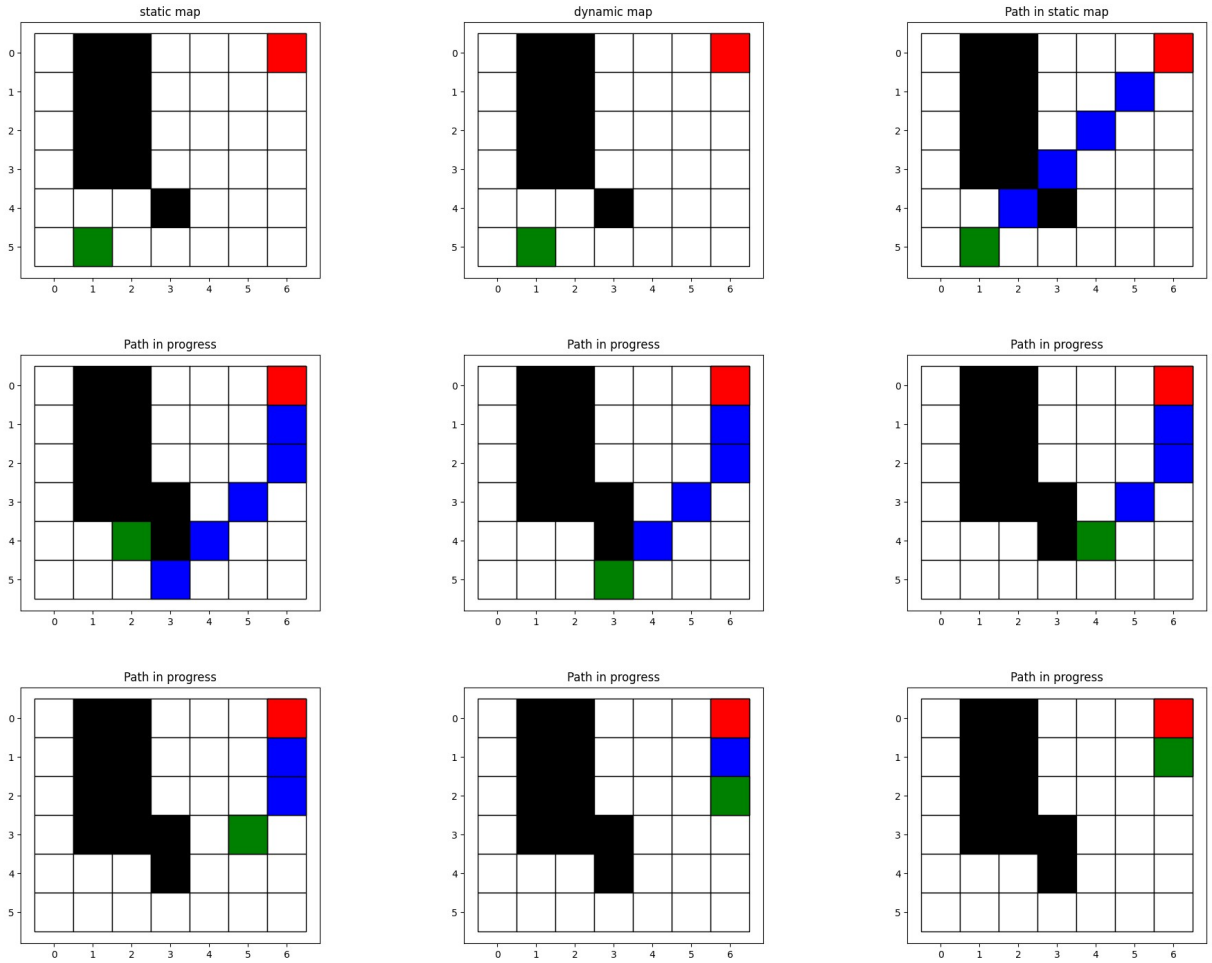
ALGORITHM IMPLEMENTATION:

D* Algorithm:

D* is also known asDynamic A* search since its similar to A*, except that it is dynamic in the sense that arc cost parameters can change during the problem-solving process. D* has two functions Process-State and Modify-Cost. Process-State computes optimal path to the goal by initially setting h(G) = 0 and inserting it into the OPEN list. The function is repeatedly called until the robot's state X is removed from the OPEN list. The Modify-Cost function changes the arc cost function, and add the affected states to the Open list. It is called if the robot detects a new obstacle.The D* algorithm is very efficient in expansive and complex environments.

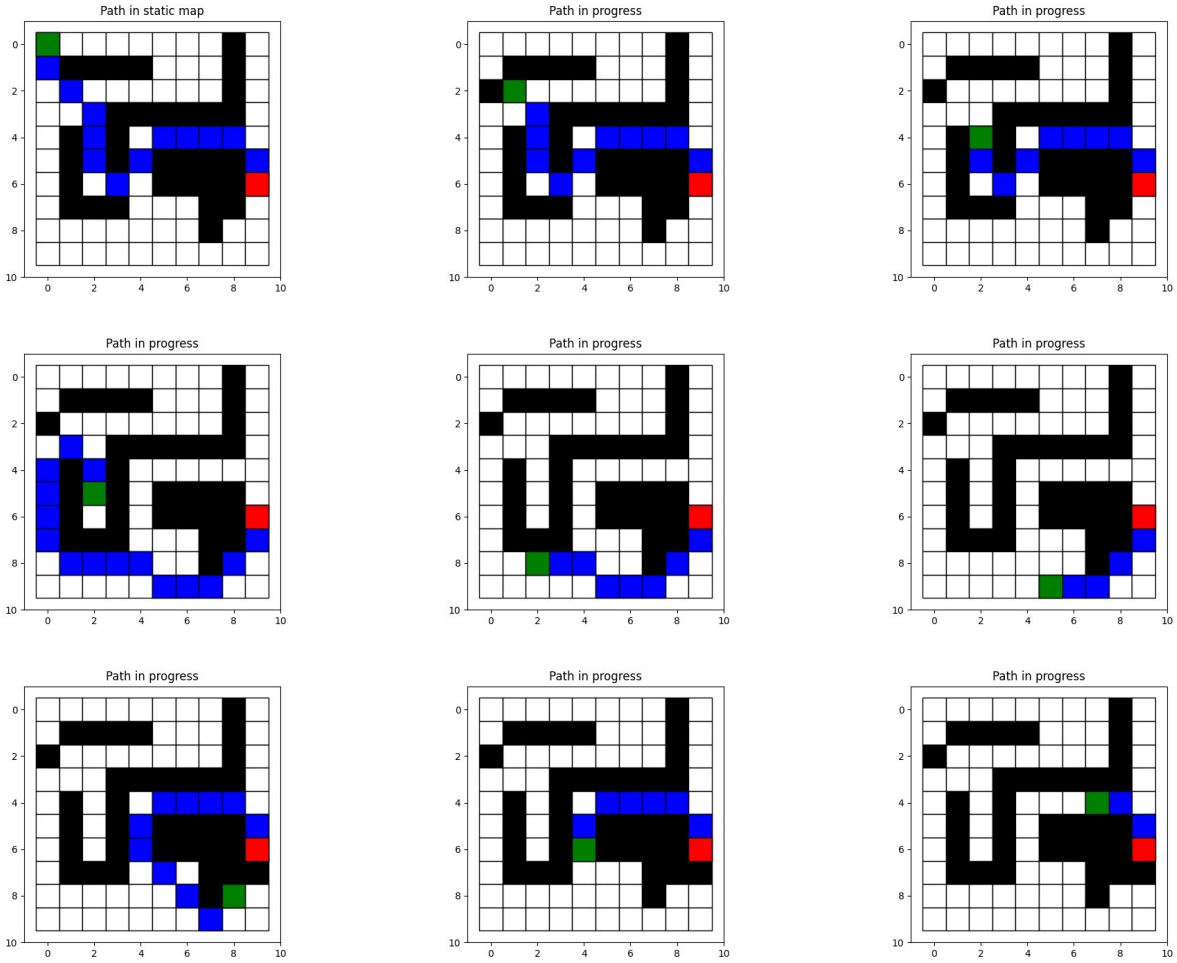
Outputs:

Scenario - 1



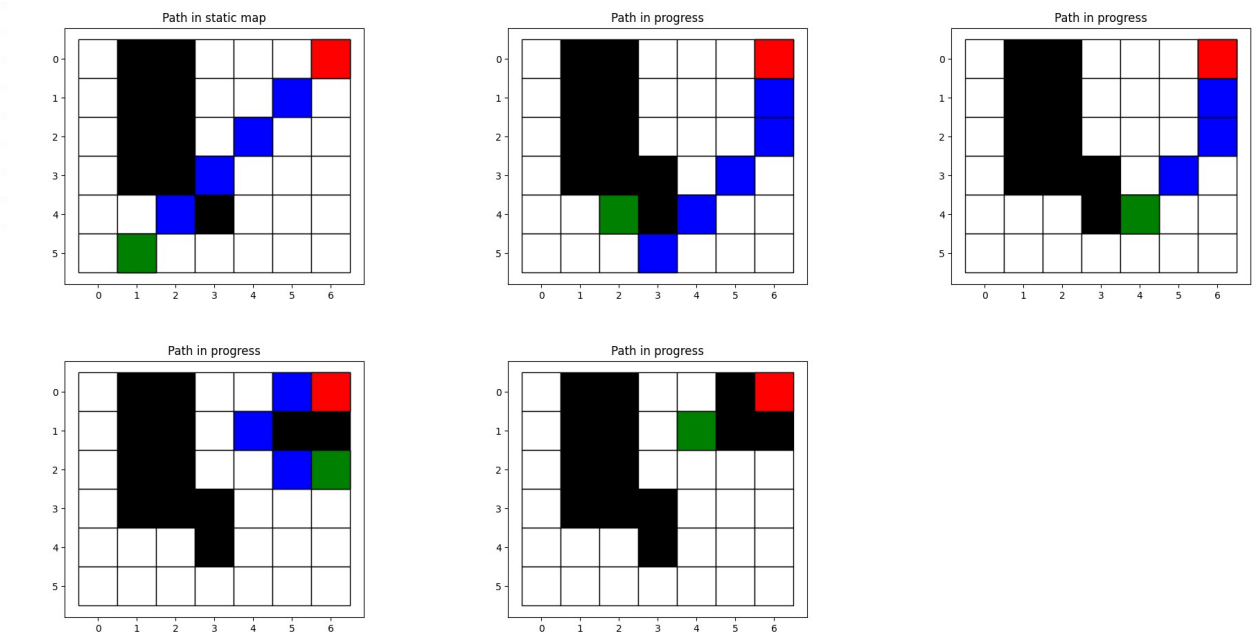
Gokul Srinivasan 500594698

Scenario - 2



Gokul Srinivasan 500594698

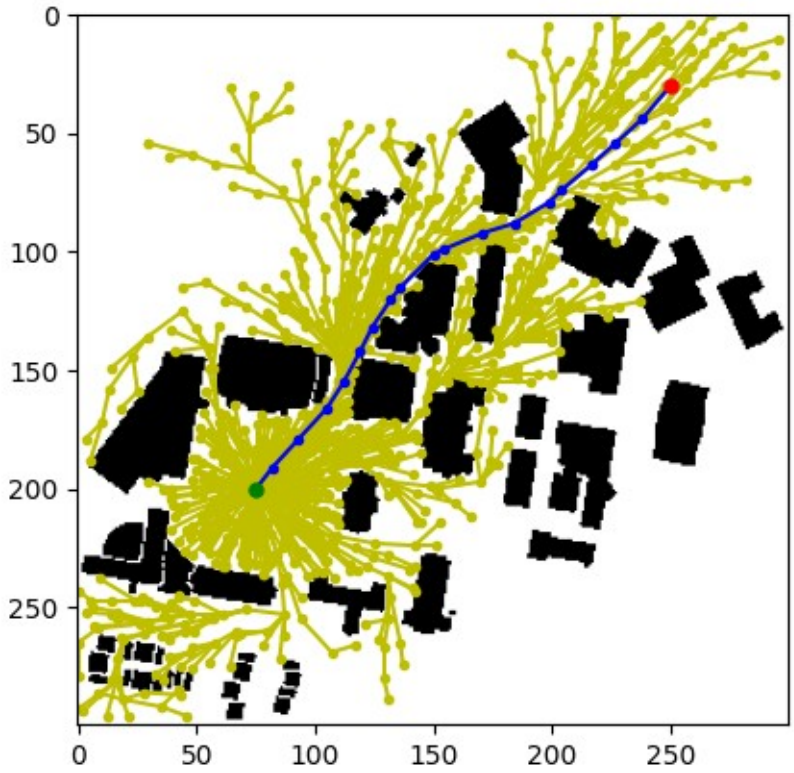
Scenario - 3



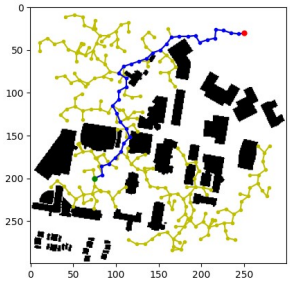
Informed RRT*

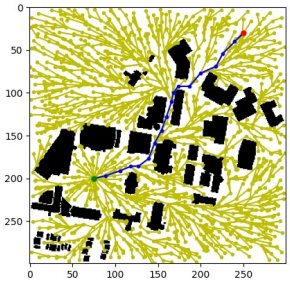
Informed RRT* is an extension of the RRT* algorithm. The main idea of informed RRT* is to focus search in a region where path is found and neglect other area. This increase the probability to get a node which reduce the path length as Sampling is random.

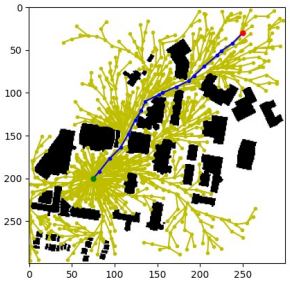
Gokul Srinivasan 500594698



Below is the comparison between RRT, RRT* and Informed RRT*. We can see that the path found by the Informed RRT* is shorter compared to the other two algorithms.







```
(cv) gs@gs:~/Desktop/WPI/Motion-Planning/Assignments/Assignment-03/informed_RRT$ python main.py
It took 251 nodes to find the current paths
The path length is 346.08
It took 1405 nodes to find the current path
The path length is 265.41
It took 1105 nodes to find the current path
The path length is 250.44
```