Joshua Peng
301340929
jpa95
CMPT 307 Coding Project:

Analysis of the three algorithms:
After running my three algorithms in separate test functions multiple times. Since we are implementing heaps, the overall running time should be around O(ElogN) for all of them. E being edges and V being nodes. But is clear to see that dijkstra's algorithm takes the longest to execute out of all the other two algorithms. This was seen when I tried to output the list of nodes visited throughout the algorithm. Especially when the starting position and the ending position were in rather far locations, the normal dijkstras algorithm essentially visits almost every single node in the graph, even if the algorithm is traversing through a node that leads to a position further than the destination point. For the A* approach however, it drastically decreased the execution time of the function as the algorithm was guiding the traverse from the starting point to a specific direction towards the destination point. When I outputted the number of nodes visited before it reaches the destination point, there was a significant decrease between A* search and Dijkstras, at some tests, the A*search would visit an average 1/3 of the nodes that Djikstra would visit. Indicating that A* search is significantly faster than Djikstra's algorithm. Lastly and the most interesting algorithm, landmark algorithm, was very similar to A* search where it finds the destination value faster than Dijkstra's. However, it really varies on the landmark points, because for mine, I used 3 landmarks and sometimes, the landmark would be put in bad positions that would significantly alter the path to the destination as it would increase the amount of visited nodes and thus effect the average of savings of the algorithm. But when there are good landmarks, the landmark algorithm visits around ¼ of the nodes that dijkstra's algorithm has to visit which is even better than A* algorithm. Therefore, at the very least, landmark algorithm still visits less nodes than Dijkstra's algorithm for good landmarks, therefore the landmark algorithm is more efficient than A* algorithm which is more efficient than Djikstra's Algorithm.