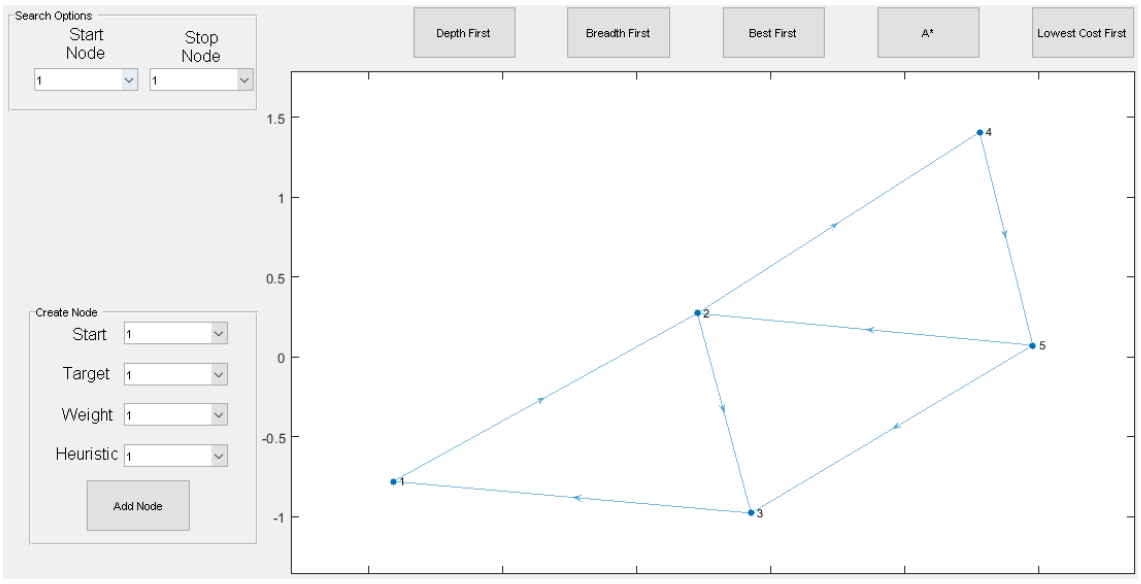
Map Navigation **Rui Carapinha**
248728

1. **Show the graph searching library you have used, briefly describe its properties.**

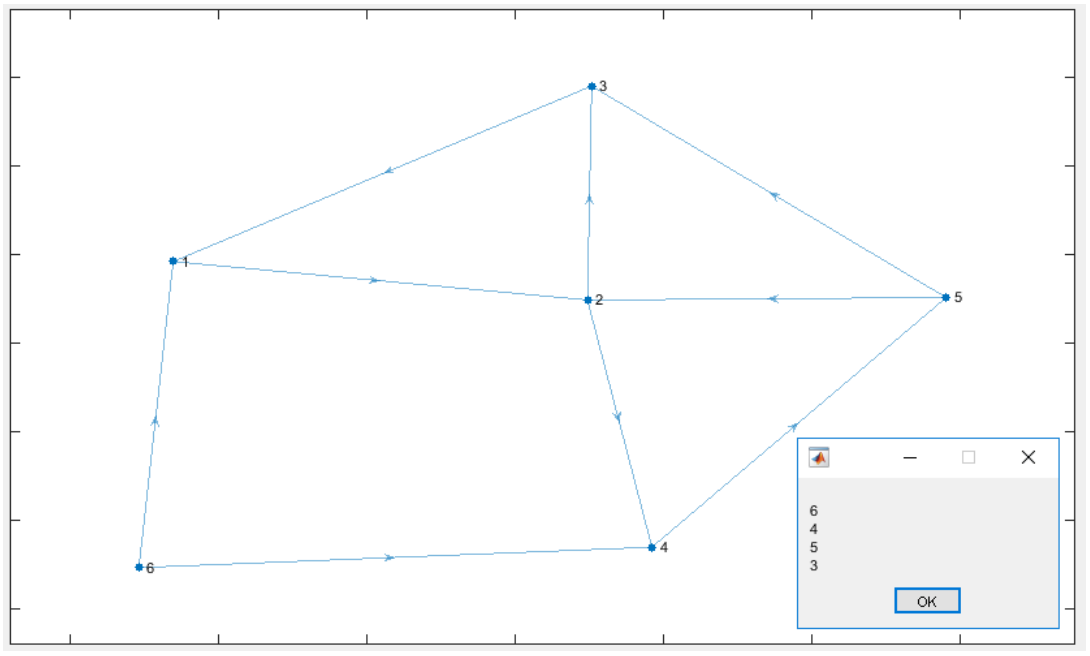
   To solve this problem, I used several functions (some directly from MATLAB and other from sources in Internet). The functions I used were:

   a. bfsearch(G,s) – this function applies breadth-first search to graph G starting at node s. The result is a vector of node IDs in order of their discovery
   b. dfsearch(G,s) – this function applies depth-first search to graph G starting at node s. The result is a vector of node IDs in order of their discovery
   c. greedybfs(source, target, weights, heuristics, StartNode, GoalNode) – this function is a slightly modified version of best-first search algorithm. The outputs of this functions give us the path the algorithm makes to get to the goal node starting in the start node.
   d. astar(source, target, weights, heuristics, startNode, goalNode) – this function implements the A* search algorithm.
   e. ucs(source, target, weights, startNode, goalNode) – this function implements a slight modified version of the lowest cost search method, called uniform cost search.

2. **Show a small example graph representation used.**



3. **Show an example solved by your program.**



```
6
4
5
3
```

Using the best-first method, starting in node 6 and finishing in node 3, the output I had was 6->4->5->3.

4. Summarize your project, describe the limitations and/or problems with your implementation

The only problems/limitations I found in my implementation were with the heuristics functions because they are constants (they don't depend on the start node and the goal node) and that's not the correct way to do it.