

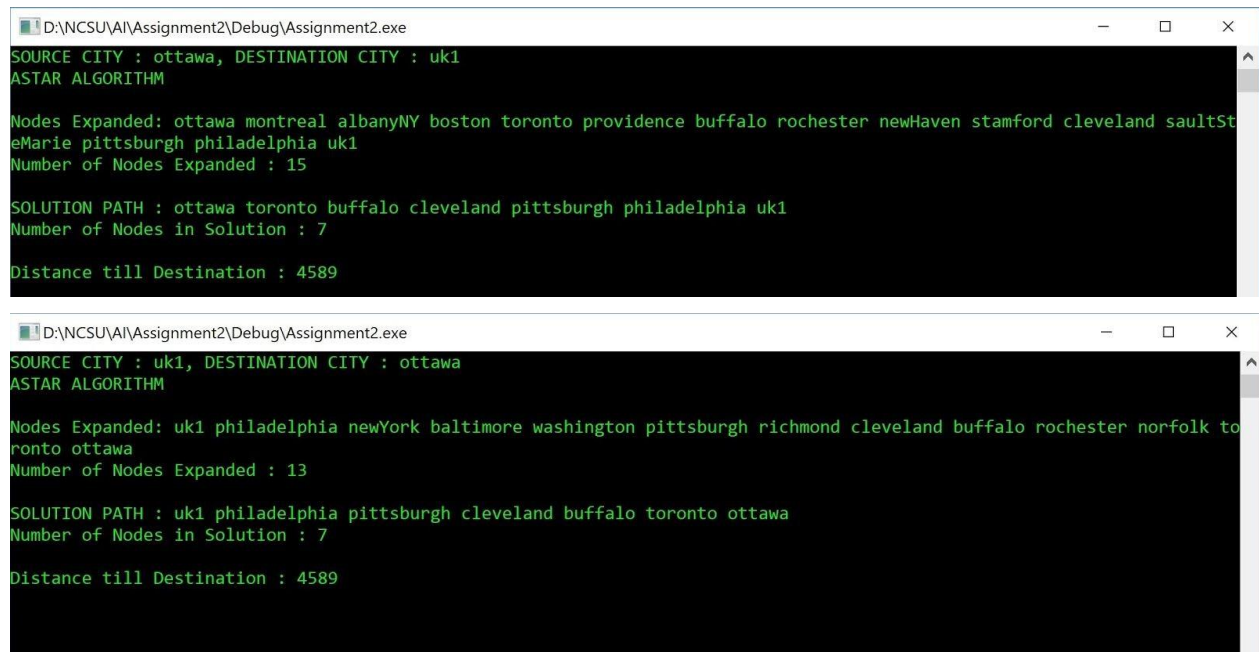
Q1.

1. Experiment with executing your implementation of A* to find various paths, until you understand the meaning of the output. Are there any pairs of cities (A,B) for which the algorithm finds a different path from B to A than from A to B? Are there any pairs of cities (A,B) for which the algorithm expands a different total number of nodes from B to A than from A to B?

SOLUTION :

No, There aren't any two cities with a path different from A to B and from B to A. A* algorithm (with an underestimating heuristic) always returns the optimal path and there can be only one optimal path in a graph containing only bi-directional edges (undirected graph).

Here is an example of a pair of city (ottawa, uk1) for which the algorithm expands a different number of nodes from ottawa to uk1 and uk1 to ottawa.



```
D:\NCSU\AI\Assignment2\Debug\Assignment2.exe
SOURCE CITY : ottawa, DESTINATION CITY : uk1
ASTAR ALGORITHM

Nodes Expanded: ottawa montreal albanynY boston toronto providence buffalo rochester newHaven stamford cleveland saultSt
eMarie pittsburgh philadelphia uk1
Number of Nodes Expanded : 15

SOLUTION PATH : ottawa toronto buffalo cleveland pittsburgh philadelphia uk1
Number of Nodes in Solution : 7

Distance till Destination : 4589

D:\NCSU\AI\Assignment2\Debug\Assignment2.exe
SOURCE CITY : uk1, DESTINATION CITY : ottawa
ASTAR ALGORITHM

Nodes Expanded: uk1 philadelphia newYork baltimore washington pittsburgh richmond cleveland buffalo rochester norfolk to
ronto ottawa
Number of Nodes Expanded : 13

SOLUTION PATH : uk1 philadelphia pittsburgh cleveland buffalo toronto ottawa
Number of Nodes in Solution : 7

Distance till Destination : 4589
```

We can see in the given output, that the number of nodes expanded from ottawa to uk1 is 15 and number of nodes expanded from uk1 to ottawa is 13.

2. Implemented the code for greedy algorithm.
3. Do enough exploration to find at least one path that is longer using greedy search than that found using A^* , or to satisfy yourself that there are no such paths. Find at least one path that is found by expanding more nodes than the comparable path using A^* , or satisfy yourself that there are no such paths. If there is such a path, list the nodes in the path and the total distance.

SOLUTION:

One path that is longer using greedy search than that found using A^* is from AUSTIN TO CHICAGO.

```

D:\NCSU\AI\Assignment2\Debug\Assignment2.exe
SOURCE CITY : austin, DESTINATION CITY : chicago
GREEDY ALGORITHM

Nodes Expanded: austin houston mexia dallas beaumont lafayette batonRouge newOrleans pensacola tallahassee albanysGA maco
n atlanta chattanooga nashville memphis littleRock tula kansasCity stLouis indianapolis cincinnati dayton columbus cleve
land pittsburgh buffalo toronto saultSteMarie thunderBay rochester wichita lincoln omaha desMoines minneapolis greenBay
milwaukee chicago
Number of Nodes Expanded : 39

SOLUTION PATH : austin houston beaumont lafayette batonRouge newOrleans pensacola tallahassee albanysGA macon atlanta cha
ttanooga nashville memphis littleRock tula kansasCity wichita lincoln omaha desMoines minneapolis greenBay milwaukee ch
icago
Number of Nodes in Solution : 25

Distance till Destination : 3738

```

```

D:\NCSU\AI\Assignment2\Debug\Assignment2.exe
SOURCE CITY : austin, DESTINATION CITY : chicago
ASTAR ALGORITHM

Nodes Expanded: austin houston sanAntonio beaumont mexia lafayette dallas batonRouge newOrleans laredo pensacola tallaha
ssee albanysGA macon atlanta chattanooga nashville lakeCity elPaso jacksonville savannah augusta albuquerque santaFe char
lotte memphis denver daytonaBeach tampa greensboro coloradoSprings orlando wichita kansasCity littleRock raleigh tucson
stLouis lincoln omaha desMoines phoenix westPalmBeach norfolk grandJunction richmond indianapolis tula mexico washington
n miami baltimore cincinnati minneapolis philadelphia dayton keyWest oklahomaCity yuma provo greenBay columbus pittsburg
h milwaukee chicago
Number of Nodes Expanded : 65

SOLUTION PATH : austin houston mexia dallas denver wichita lincoln omaha desMoines minneapolis greenBay milwaukee chicag
o
Number of Nodes in Solution : 13

Distance till Destination : 2976

```

We see that A^* algorithm gives a path with 13 nodes while Greedy algorithm gives a path with 25 nodes. Also, the path cost for A^* is much less than the cost for greedy algorithm.

One path that is found expanding more nodes in Greedy Algorithm than the comparable path using A^* is from ATLANTA to AUSTIN

We see that the number of nodes expanded in A^* algorithm is 15 while greedy algorithm expands 18 nodes.

```
D:\NCSU\AI\Assignment2\Debug\Assignment2.exe
SOURCE CITY : atlanta, DESTINATION CITY : austin
GREEDY ALGORITHM

Nodes Expanded: atlanta chattanooga nashville memphis littleRock tula oklahomaCity ftWorth kansasCity wichita stLouis l
lincoln omaha denver dallas mexia houston austin
Number of Nodes Expanded : 18

SOLUTION PATH : atlanta chattanooga nashville memphis littleRock tula kansasCity wichita denver dallas mexia houston au
stin
Number of Nodes in Solution : 13

Distance till Destination : 3057

D:\NCSU\AI\Assignment2\Debug\Assignment2.exe
SOURCE CITY : atlanta, DESTINATION CITY : austin
ASTAR ALGORITHM

Nodes Expanded: atlanta chattanooga macon albanyGA nashville memphis littleRock tallahassee pensacola newOrleans batonRo
uge lafayette beaumont houston austin
Number of Nodes Expanded : 15

SOLUTION PATH : atlanta macon albanyGA tallahassee pensacola newOrleans batonRouge lafayette beaumont houston austin
Number of Nodes in Solution : 11

Distance till Destination : 1203
```

4. [10 points] Change your code so as to implement uniform cost search, as discussed in the web notes.

Implemented the Uniform Cost Search Algorithm.

5. Do enough exploration to find at least one path that is longer using uniform cost than that found using A^* , or to satisfy yourself that there are no such paths. Find at least one path that is found by expanding more nodes than the comparable path using A^* , or satisfy yourself that there are no such paths. If there is such a path, list the nodes in the path and the total distance.

/*

One path that is longer using uniform cost than that found using A^* : There is **no path** for which A^* algorithm gives a shorter path than uniform cost algorithm. We know Uniform Cost always gives the least cost path and since, there was no path with a different size than Uniform Cost when run on A^* implies that the heuristic used for A^* is underestimating and hence, A^* also gives the optimal path. This is because both the algorithm guarantee shortest path(given heuristic is underestimating).

*/

One path that is found by expanding more nodes in Uniform Cost Search than the comparable path using A^* is from

```
D:\NCSU\AI\Assignment2\Debug\Assignment2.exe
SOURCE CITY : calgary, DESTINATION CITY : elPaso
ASTAR ALGORITHM

Nodes Expanded: calgary vancouver seattle winnipeg portland salem eugene boise medford saltLakeCity redding minneapolis
desMoines omaha lincoln pointReyes sacramento stockton modesto wichita fresno reno bakersfield thunderBay sanFrancisco o
akland lasVegas sanJose losAngeles salinas sanDiego sanLuisObispo yuma phoenix tucson greenBay elPaso
Number of Nodes Expanded : 37

SOLUTION PATH : calgary vancouver seattle portland salem eugene medford redding pointReyes sacramento stockton modesto f
resno bakersfield losAngeles sanDiego yuma phoenix tucson elPaso
Number of Nodes in Solution : 20

Distance till Destination : 2968

D:\NCSU\AI\Assignment2\Debug\Assignment2.exe
SOURCE CITY : calgary, DESTINATION CITY : elPaso
UNIFORM ALGORITHM

Nodes Expanded: calgary vancouver seattle winnipeg portland salem eugene medford thunderBay minneapolis redding boise po
intReyes desMoines greenBay sacramento saltLakeCity omaha stockton saultSteMarie milwaukee modesto lincoln sanFrancisco
oakland reno sanJose chicago salinas fresno bakersfield sanLuisObispo wichita losAngeles midland toronto lasVegas toledo
sanDiego kansasCity buffalo rochester yuma ottawa cleveland tula stLouis albanyNY phoenix denver montreal oklahomaCity
columbus pittsburgh coloradoSprings boston tucson dayton providence indianapolis cincinnati littleRock ftWorth grandJun
ction newHaven memphis stamford santaFe philadelphia elPaso
Number of Nodes Expanded : 70

SOLUTION PATH : calgary vancouver seattle portland salem eugene medford redding pointReyes sacramento stockton modesto f
resno bakersfield losAngeles sanDiego yuma phoenix tucson elPaso
Number of Nodes in Solution : 20

Distance till Destination : 2968
```

We see that the number of nodes expanded in A* algorithm is 37 while Uniform Cost search expands 70 nodes.

6. As part of your answer, compare the solution paths and explain what happened, especially any weird behavior you might detect.

The A* algorithm always gives the optimal solution if the heuristic is underestimating. Hence, it does not matter for a path from A to B and from B to A, as the path remains the same which is optimal i.e. with the least path cost. Initially, we don't know whether the heuristic is underestimating. But when we find all the paths found by Astar equal to the paths found by uniform cost, it was confirmed that the heuristic is underestimating. By comparing Greedy Search algorithm with A* algorithm, we see that Greedy Search algorithm does not guarantee the optimal path. Although, it does expand lesser number of nodes to reach the target city, it would not mind going on a path more costly. Greedy algorithm is thus much faster than A* and also has less memory usage in terms of expanded nodes storage but will not return an optimal path.

By comparing Greedy Search algorithm with A* algorithm, we don't find any two cities with different solutions confirming the heuristic being underestimating. I literally ran the code for all the possible city combinations and failed to find such a case. Uniform Cost does use a lot more memory in terms of nodes expanded than A* algorithm in most cases.