Dana Luna (260857641)
COMP424 - Assignment 1

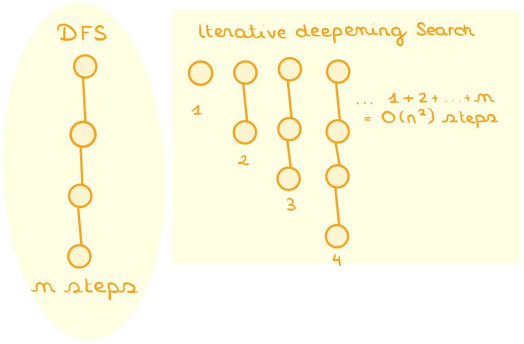
Question 1

b) Manhattan distance is always smaller or equal to the true number of movements. The cost of each step in the Manhattan distance heuristic is 1 and one is smaller or equal than a is the set {1,2,3,4,5,6,7}. Hence
(Manhattan distance * 1) <= (true number of moves * a)
Therefore, ***h(n) <= h*(n)*** by the definition of an admissible heuristic and ***h(n)*** is admissible
Since our heuristic is admissible, we can now do A* search with this h(n).

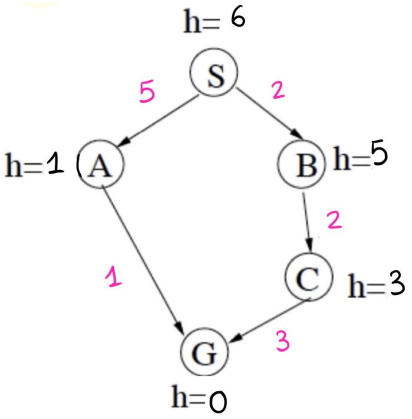
c) We could use the Manhattan distance multiplied by the number on the piece moved.

Question 2

a) Yes, If every state has only one child, then DFS will find the answer in n steps which is ***O(n)*** comparing to ***O(n^2)*** for Iterative Deepening Search.



b) Yes, Breadth-first search is a special case of the uniform-cost search when the step costs are equal. In this case uniform-cost search is equal to Breadth-first search.
c) Yes, when the heuristic function is a constant 0 function, ***h(n)=0***



d) No, the algorithm will expand the nodes {S,A,G} since h(A) < h(B) but the path {S,B,C,G} is actually shorter (see counter example)

e) Yes**,** as the optimal solution is unique and the heuristic is perfect, A* expand nodes with the best heuristic. It will therefore expand the nodes on the optimal path since there is no tie and the heuristic is optimal.

Output of question 1 a)

Using BFS to solve 6 puzzle problem...
1 4 2
5 3 0
|
v
1 4 2

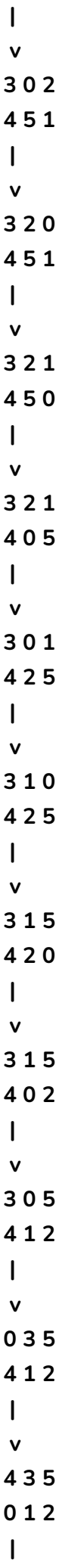
5 0 3
|
v
1 0 2
5 4 3
|
v
0 1 2
5 4 3
|
v
PATH LENGTH IS 4

Using UNIFORM COST SEARCH to solve 6 puzzle problem...
1 4 2
5 3 0
|
v
1 4 2
5 0 3
|
v
1 0 2
5 4 3
|
v
0 1 2
5 4 3
|
v
PATH LENGTH IS 4

Using BFS to solve 6 puzzle problem...
1 4 2
5 3 0
|
v
1 4 0
5 3 2
|
v
1 0 4
5 3 2
|
v

0 1 4
5 3 2
|
v
5 1 4
0 3 2
|
v
5 1 4
3 0 2
|
v
5 0 4
3 1 2
|
v
5 4 0
3 1 2
|
v
5 4 2
3 1 0
|
v
5 4 2
3 0 1
|
v
5 4 2
0 3 1
|
v
0 4 2
5 3 1
|
v
4 0 2
5 3 1
|
v
4 2 0
5 3 1
|
v
4 2 1

5 3 0
|
v
4 2 1
5 0 3
|
v
4 0 1
5 2 3
|
v
4 1 0
5 2 3
|
v
4 1 3
5 2 0
|
v
4 1 3
5 0 2
|
v
4 0 3
5 1 2
|
v
4 3 0
5 1 2
|
v
4 3 2
5 1 0
|
v
4 3 2
5 0 1
|
v
4 3 2
0 5 1
|
v
0 3 2
4 5 1


3 0 2
4 5 1
3 2 0
4 5 1
3 2 1
4 5 0
3 2 1
4 0 5
3 0 1
4 2 5
3 1 0
4 2 5
3 1 5
4 2 0
3 1 5
4 0 2
3 0 5
4 1 2
0 3 5
4 1 2
4 3 5
0 1 2

v
4 3 5
1 0 2
|
v
4 3 5
1 2 0
|
v
4 3 0
1 2 5
|
v
4 0 3
1 2 5
|
v
4 2 3
1 0 5
|
v
4 2 3
0 1 5
|
v
0 2 3
4 1 5
|
v
2 0 3
4 1 5
|
v
2 1 3
4 0 5
|
v
2 1 3
0 4 5
|
v
0 1 3
2 4 5
|
v

1 0 3
2 4 5
|
v
1 3 0
2 4 5
|
v
1 3 5
2 4 0
|
v
1 3 5
2 0 4
|
v
1 3 5
0 2 4
|
v
0 3 5
1 2 4
|
v
3 0 5
1 2 4
|
v
3 2 5
1 0 4
|
v
3 2 5
0 1 4
|
v
0 2 5
3 1 4
|
v
2 0 5
3 1 4
|
v
2 1 5

3 0 4
|
v
2 1 5
0 3 4
|
v
0 1 5
2 3 4
|
v
1 0 5
2 3 4
|
v
1 5 0
2 3 4
|
v
1 5 4
2 3 0
|
v
1 5 4
2 0 3
|
v
1 5 4
0 2 3
|
v
0 5 4
1 2 3
|
v
5 0 4
1 2 3
|
v
5 2 4
1 0 3
|
v
5 2 4
0 1 3

|
v
0 2 4
5 1 3
|
v
2 0 4
5 1 3
|
v
2 1 4
5 0 3
|
v
2 1 4
5 3 0
|
v
2 1 0
5 3 4
|
v
2 0 1
5 3 4
|
v
0 2 1
5 3 4
|
v
5 2 1
0 3 4
|
v
5 2 1
3 0 4
|
v
5 0 1
3 2 4
|
v
0 5 1
3 2 4
|

v
3 5 1
0 2 4
|
v
3 5 1
2 0 4
|
v
3 5 1
2 4 0
|
v
3 5 0
2 4 1
|
v
3 0 5
2 4 1
|
v
0 3 5
2 4 1
|
v
2 3 5
0 4 1
|
v
2 3 5
4 0 1
|
v
2 0 5
4 3 1
|
v
0 2 5
4 3 1
|
v
4 2 5
0 3 1
|
v

4 2 5
3 0 1
|
v
4 2 5
3 1 0
|
v
4 2 0
3 1 5
|
v
4 0 2
3 1 5
|
v
4 1 2
3 0 5
|
v
4 1 2
0 3 5
|
v
0 1 2
4 3 5
|
v
1 0 2
4 3 5
|
v
1 2 0
4 3 5
|
v
1 2 5
4 3 0
|
v
1 2 5
4 0 3
|
v
1 0 5

4 2 3
|
v
0 1 5
4 2 3
|
v
4 1 5
0 2 3
|
v
4 1 5
2 0 3
|
v
4 0 5
2 1 3
|
v
0 4 5
2 1 3
|
v
2 4 5
0 1 3
|
v
2 4 5
1 0 3
|
v
2 4 5
1 3 0
|
v
2 4 0
1 3 5
|
v
2 0 4
1 3 5
|
v
0 2 4
1 3 5

|
v
1 2 4
0 3 5
|
v
1 2 4
3 0 5
|
v
1 0 4
3 2 5
|
v
0 1 4
3 2 5
|
v
3 1 4
0 2 5
|
v
3 1 4
2 0 5
|
v
3 0 4
2 1 5
|
v
0 3 4
2 1 5
|
v
2 3 4
0 1 5
|
v
2 3 4
1 0 5
|
v
2 3 4
1 5 0
|

v
2 3 0
1 5 4
|
v
2 0 3
1 5 4
|
v
0 2 3
1 5 4
|
v
1 2 3
0 5 4
|
v
1 2 3
5 0 4
|
v
1 0 3
5 2 4
|
v
0 1 3
5 2 4
|
v
5 1 3
0 2 4
|
v
5 1 3
2 0 4
|
v
5 0 3
2 1 4
|
v
5 3 0
2 1 4
|
v

5 3 4
2 1 0
|
v
5 3 4
2 0 1
|
v
5 3 4
0 2 1
|
v
0 3 4
5 2 1
|
v
3 0 4
5 2 1
|
v
3 2 4
5 0 1
|
v
3 2 4
5 1 0
|
v
3 2 0
5 1 4
|
v
3 0 2
5 1 4
|
v
3 1 2
5 0 4
|
v
3 1 2
5 4 0
|
v
3 1 0

5 4 2
|
v
3 0 1
5 4 2
|
v
0 3 1
5 4 2
|
v
5 3 1
0 4 2
|
v
5 3 1
4 0 2
|
v
5 3 1
4 2 0
|
v
5 3 0
4 2 1
|
v
5 0 3
4 2 1
|
v
5 2 3
4 0 1
|
v
5 2 3
4 1 0
|
v
5 2 0
4 1 3
|
v
5 0 2
4 1 3

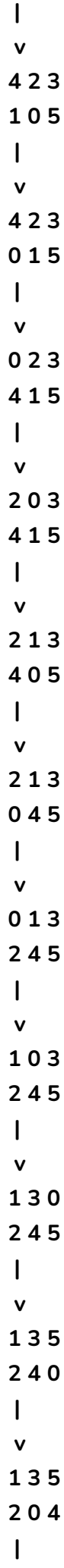
|
v
5 1 2
4 0 3
|
v
5 1 2
0 4 3
|
v
0 1 2
5 4 3
|
v
PATH LENGTH IS 164

IDDFS : Found a solution at level : 4
Using Iterative Deepening (IDDFS) to solve 6 puzzle problem...
1 4 2
5 3 0
|
v
1 4 0
5 3 2
|
v
1 0 4
5 3 2
|
v
0 1 4
5 3 2
|
v
5 1 4
0 3 2
|
v
5 1 4
3 0 2
|
v
5 0 4
3 1 2
|

v
5 4 0
3 1 2
|
v
5 4 2
3 1 0
|
v
5 4 2
3 0 1
|
v
5 4 2
0 3 1
|
v
0 4 2
5 3 1
|
v
4 0 2
5 3 1
|
v
4 2 0
5 3 1
|
v
4 2 1
5 3 0
|
v
4 2 1
5 0 3
|
v
4 0 1
5 2 3
|
v
4 1 0
5 2 3
|
v

4 1 3
5 2 0
|
v
4 1 3
5 0 2
|
v
4 0 3
5 1 2
|
v
4 3 0
5 1 2
|
v
4 3 2
5 1 0
|
v
4 3 2
5 0 1
|
v
4 3 2
0 5 1
|
v
0 3 2
4 5 1
|
v
3 0 2
4 5 1
|
v
3 2 0
4 5 1
|
v
3 2 1
4 5 0
|
v
3 2 1

4 0 5
|
v
3 0 1
4 2 5
|
v
3 1 0
4 2 5
|
v
3 1 5
4 2 0
|
v
3 1 5
4 0 2
|
v
3 0 5
4 1 2
|
v
0 3 5
4 1 2
|
v
4 3 5
0 1 2
|
v
4 3 5
1 0 2
|
v
4 3 5
1 2 0
|
v
4 3 0
1 2 5
|
v
4 0 3
1 2 5



v
1 3 5
0 2 4
|
v
0 3 5
1 2 4
|
v
3 0 5
1 2 4
|
v
3 2 5
1 0 4
|
v
3 2 5
0 1 4
|
v
0 2 5
3 1 4
|
v
2 0 5
3 1 4
|
v
2 1 5
3 0 4
|
v
2 1 5
0 3 4
|
v
0 1 5
2 3 4
|
v
1 0 5
2 3 4
|
v

1 5 0
2 3 4
|
v
1 5 4
2 3 0
|
v
1 5 4
2 0 3
|
v
1 5 4
0 2 3
|
v
0 5 4
1 2 3
|
v
5 0 4
1 2 3
|
v
5 2 4
1 0 3
|
v
5 2 4
0 1 3
|
v
0 2 4
5 1 3
|
v
2 0 4
5 1 3
|
v
2 1 4
5 0 3
|
v
2 1 4

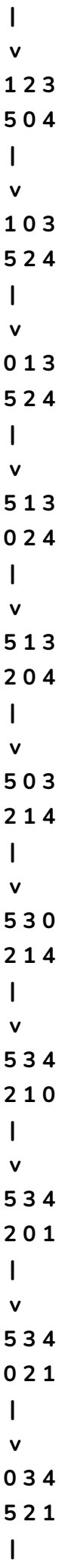
5 3 0
|
v
2 1 0
5 3 4
|
v
2 0 1
5 3 4
|
v
0 2 1
5 3 4
|
v
5 2 1
0 3 4
|
v
5 2 1
3 0 4
|
v
5 0 1
3 2 4
|
v
0 5 1
3 2 4
|
v
3 5 1
0 2 4
|
v
3 5 1
2 0 4
|
v
3 5 1
2 4 0
|
v
3 5 0
2 4 1

|
v
3 0 5
2 4 1
|
v
0 3 5
2 4 1
|
v
2 3 5
0 4 1
|
v
2 3 5
4 0 1
|
v
2 0 5
4 3 1
|
v
0 2 5
4 3 1
|
v
4 2 5
0 3 1
|
v
4 2 5
3 0 1
|
v
4 2 5
3 1 0
|
v
4 2 0
3 1 5
|
v
4 0 2
3 1 5
|

v
4 1 2
3 0 5
|
v
4 1 2
0 3 5
|
v
0 1 2
4 3 5
|
v
1 0 2
4 3 5
|
v
1 2 0
4 3 5
|
v
1 2 5
4 3 0
|
v
1 2 5
4 0 3
|
v
1 0 5
4 2 3
|
v
0 1 5
4 2 3
|
v
4 1 5
0 2 3
|
v
4 1 5
2 0 3
|
v

4 0 5
2 1 3
|
v
0 4 5
2 1 3
|
v
2 4 5
0 1 3
|
v
2 4 5
1 0 3
|
v
2 4 5
1 3 0
|
v
2 4 0
1 3 5
|
v
2 0 4
1 3 5
|
v
0 2 4
1 3 5
|
v
1 2 4
0 3 5
|
v
1 2 4
3 0 5
|
v
1 0 4
3 2 5
|
v
0 1 4

3 2 5
|
v
3 1 4
0 2 5
|
v
3 1 4
2 0 5
|
v
3 0 4
2 1 5
|
v
0 3 4
2 1 5
|
v
2 3 4
0 1 5
|
v
2 3 4
1 0 5
|
v
2 3 4
1 5 0
|
v
2 3 0
1 5 4
|
v
2 0 3
1 5 4
|
v
0 2 3
1 5 4
|
v
1 2 3
0 5 4


1 2 3
5 0 4
1 0 3
5 2 4
0 1 3
5 2 4
5 1 3
0 2 4
5 1 3
2 0 4
5 0 3
2 1 4
5 3 0
2 1 4
5 3 4
2 1 0
5 3 4
2 0 1
5 3 4
0 2 1
0 3 4
5 2 1

v
3 0 4
5 2 1
|
v
3 2 4
5 0 1
|
v
3 2 4
5 1 0
|
v
3 2 0
5 1 4
|
v
3 0 2
5 1 4
|
v
3 1 2
5 0 4
|
v
3 1 2
5 4 0
|
v
3 1 0
5 4 2
|
v
3 0 1
5 4 2
|
v
0 3 1
5 4 2
|
v
5 3 1
0 4 2
|
v

5 3 1
4 0 2
|
v
5 3 1
4 2 0
|
v
5 3 0
4 2 1
|
v
5 0 3
4 2 1
|
v
5 2 3
4 0 1
|
v
5 2 3
4 1 0
|
v
5 2 0
4 1 3
|
v
5 0 2
4 1 3
|
v
5 1 2
4 0 3
|
v
5 1 2
0 4 3
|
v
0 1 2
5 4 3
|
v
PATH LENGTH IS 164

Ouput for question 3 (run TSP)

Question 3 (a) :
Min : 1.4079764682679343
Max : 3.1906389562444923
Mean : 2.4794810044247853
SD : 0.3285348973010342

Question 3 (b) :
Min : 2.0137183129283756
Max : 5.238865414507833
Mean : 3.6513322433824986
SD : 0.712234671273446
Optimal solutions : 0

Question 3 (c) :
Min : 1.4079764682679345
Max : 3.257136958446046
Mean : 2.48662990516553
SD : 0.3338387255025535
Optimal solutions : 65

Question 3 (b) with 100 cities:
Min : 45.85235117523879
Max : 59.289193675312106
Mean : 51.91580859902387
SD : 2.8525107966743195

Question 3 (c) (b) with 100 cities:
Min : 7.5218387326519 79
Max : 9.099059290804252
Mean : 8.24059919970573
SD : 0.29634181217469874
