Programming Assignment - I
6160 Advanced Topics in Artificial Intelligence

Maruf Aytekin
aaytekin@gmail.com

March 18, 2015

1 Problem Definition

Write a program to solve the 8-puzzle problem using the following search algorithms; A* search, breadth-first search, uniform-cost search, depth-first search, iterative deepening search, and greedy best search.

2 Data Set

Following 8-puzzles are selected which require less than 15 levels of depth to be resolved. Input states:

1	0	3	1	2	3	1	2	3	1	3	0	4	1	3	1	2	3	2	3	5	2	3	5	2	3	5
4	2	5	4	0	5	0	4	5	4	2	5	0	2	6	0	7	6	1	0	4	1	4	0	1	0	4
7	8	6	7	8	6	7	8	6	7	8	6	7	5	8	5	4	8	7	8	6	7	8	6	7	8	6

The Goal state:

1	2	3
4	5	6
7	8	0

3 Results

		Puzzles (1-10)										Median
A* Search	Expanded	4	3	2	3	4	5	14	9	8	9	6.1
	In Memory	9	9	9	9	9	11	23	23	23	23	14.8
Depth First Search (15)	Expanded	36726	15	5965	13708	58475	12076	16963	92145	172918	13956	42294.7
	In Memory	25	25	25	25	25	25	25	28	28	28	25.9
Breadth First Search	Expanded	32	14	9	11	35	53	160	447	210	451	142.2
	In Memory	27	15	11	13	29	47	129	283	152	286	99.2
Uniform Cost Search	Expanded	24	10	6	18	37	66	205	464	227	443	150.0
	In Memory	29	29	29	29	33	62	158	307	307	307	129.0
Greedy Best Search	Expanded	4	3	2	3	4	5	7	8	7	8	5.1
	In Memory	9	9	9	9	9	11	16	17	17	17	12.3
Iterative Deepening Search	Expanded	36	25	13	40	20	11	13	450	2461	1879	494.8
	In Memory	8	8	8	8	8	9	15	15	16	16	11.1

4 Conclusion

In terms of expended node numbers A* search and greedy best search seems performed better when they are compared against the others. However, greedy best search gets lost in loops or unable to find a solution. The data set is chosen from simple puzzles to make sure greedy search does not get lost.

During the tests we limited depth first search(DFS)'s maximum depth to 15 because DFS algorithm gets lost or runs into endless loops which causes out of memory exception most of the time.