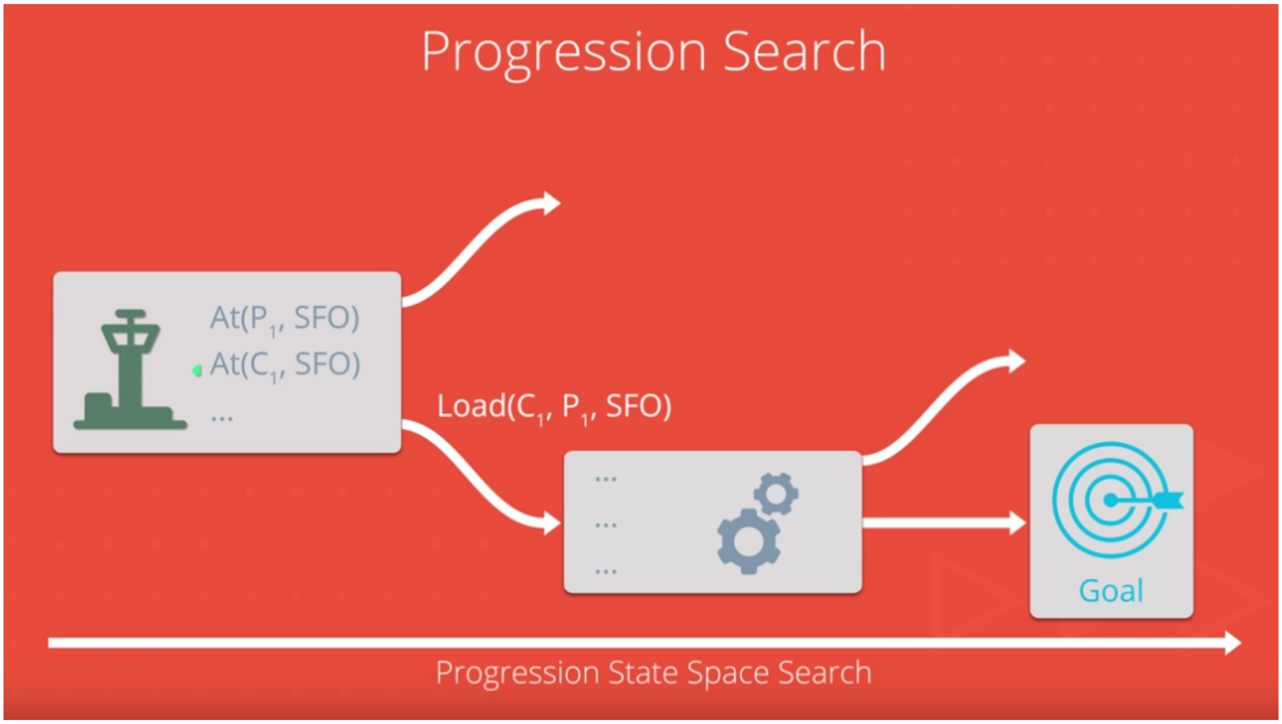
HEURISTIC ANALYSIS FOR IMPLEMENTING A PLANNING SEARCH AGENT

Tejasvi Nuthalapati
As part of Artificial Intelligence Nano Degree Program at Udacity



INDEX OF CONTENTS

1. SYNOPSIS

In this project I implemented a planning search agent to solve **deterministic logistics** planning problem for an **Air Cargo Transport system.**

All problems in the Air Cargo Domain use this, **Action Schema:**

```
Action(Load(c, p, a),
            PRECOND: At(c, a) ∧ At(p, a) ∧ Cargo(c) ∧ Plane(p) ∧ Airport(a)
            EFFECT: ¬ At(c, a) ∧ In(c, p))
Action(Unload(c, p, a),
            PRECOND: In(c, p) ∧ At(p, a) ∧ Cargo(c) ∧ Plane(p) ∧ Airport(a)
            EFFECT: At(c, a) ∧ ¬ In(c, p))
Action(Fly(p, from, to),
            PRECOND: At(p, from) ∧ Plane(p) ∧ Airport(from) ∧ Airport(to)
            EFFECT: ¬ At(p, from) ∧ At(p, to))
```

There are three Problems that are given in this Action Schema

SNO	Initial state and Goals
Problem-1	Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(P1, SFO) ∧ At(P2, JFK) ∧ Cargo(C1) ∧ Cargo(C2) ∧ Plane(P1) ∧ Plane(P2) ∧ Airport(JFK) ∧ Airport(SFO)) Goal(At(C1, JFK) ∧ At(C2, SFO))
Problem-2	Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(C3, ATL) ∧ At(P1, SFO) ∧ At(P2, JFK) ∧ At(P3, ATL) ∧ Cargo(C1) ∧ Cargo(C2) ∧ Cargo(C3) ∧ Plane(P1) ∧ Plane(P2) ∧ Plane(P3) ∧ Airport(JFK) ∧ Airport(SFO) ∧ Airport(ATL)) Goal(At(C1, JFK) ∧ At(C2, SFO) ∧ At(C3, SFO))
Problem-3	Init(At(C1, SFO) ∧ At(C2, JFK) ∧ At(C3, ATL) ∧ At(C4, ORD) ∧ At(P1, SFO) ∧ At(P2, JFK) ∧ Cargo(C1) ∧ Cargo(C2) ∧ Cargo(C3) ∧ Cargo(C4) ∧ Plane(P1) ∧ Plane(P2) ∧ Airport(JFK) ∧ Airport(SFO) ∧ Airport(ATL) ∧ Airport(ORD)) Goal(At(C1, JFK) ∧ At(C3, JFK) ∧ At(C2, SFO) ∧ At(C4, SFO))

And their optimal Sequence of actions identified are:

SNO	Optimal Sequence of Actions

Problem-1	Plan Length - 6
	Load(C1,P1,SFO)
	Load(C2,P2,JFK)
	Fly(P2,JFK,SFO)
	UnLoad(C2,P2,SFO)
	Fly(P1,SFO,JFK)
	UnLoad(C2,P1,JFK)

Problem-2	Plan Length - 9
	Load(C1,P1,SFO)
	Load(C2,P2,JFK)
	Load(C3,P3,ATL)
	Fly(P2,JFK,SFO)
	UnLoad(C2,P2,SFO)
	Fly(P1,SFO,JFK)
	UnLoad(C1,P1,JFK)
	Fly(P3,ATL,SFO)
	UnLoad(C3,P3,SFO)

Problem-3	Plan Length - 12
	Load(C1,P1,SFO)
	Load(C2,P2,JFK)
	Fly(P1,SFO,ATL)
	Load(C3,P1,ATL)
	Fly(P2,JFK,ORD)
	Load(C4,P2,ORD)
	Fly(P2,ORD,SFO)
	Fly(P1,ATL,JFK)
	UnLoad(C4,P2,SFO)
	UnLoad(C3,P1,JFK)
	UnLoad(C2,P2,SFO)
	UnLoad(C1,P1,JFK)

2. ANALYSIS

2.1 Un-Informed Search Strategies:

Uninformed search strategies use only the information available in the problem definition, the strategies are: Breadth First Search, Uniform-cost search, Depth-first search, Depth-limited search and Iterative deepening search.

Problem 1 Results:

Strategy	**Optimal**	**Path Length**	**Time(sec)**	**Node Expansions**
BFS	**Yes**	**6**	0.03	43
BF-Tree Search	**Yes**	**6**	0.967	1458
DF-Graph Search	No	20	**0.014**	21
Depth Limited Search	No	50	0.09	101
Uniform Cost Search	**Yes**	**6**	0.04	55
Recursive Best First Search	**Yes**	**6**	2.81	4229
Greedy Best First Search	**Yes**	**6**	**0.005**	7

Problem 2 Results:

Strategy	**Optimal**	**Path Length**	**Time(sec)**	**Node Expansions**
BFS	**Yes**	**9**	14.10	3343
BF-Tree Search	-	-	-	-
DF-Graph Search	No	619	**3.5175**	**624**
Depth Limited Search	No	50	945.28	222719
Uniform Cost Search	**Yes**	**9**	12.51	4852
Recursive Best First Search	-	-	-	-
Greedy Best First Search	**Yes**	**21**	**2.56**	990

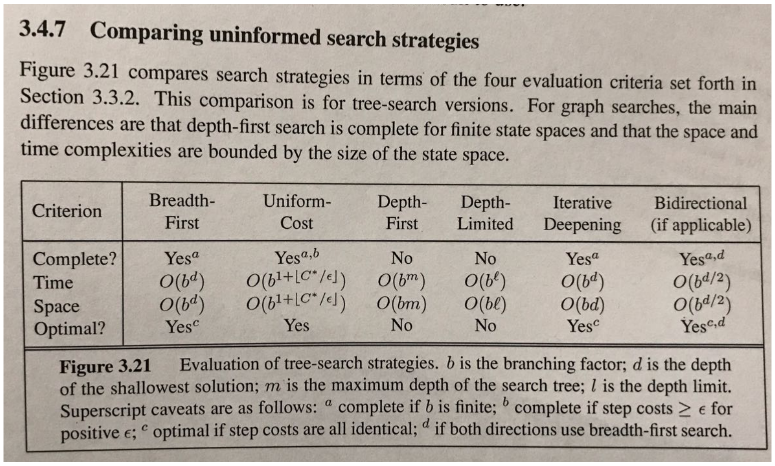
Problem 3 Results:

Strategy	**Optimal**	**Path Length**	**Time(sec)**	**Node Expansions**
BFS	**Yes**	**12**	117.22	14663
BF-Tree Search	-	-	-	-
DF-Graph Search	No	392	**2.163**	**408**
Depth Limited Search	-	-	-	-
Uniform Cost Search	**Yes**	**12**	62.028	18236

Recursive Best First Search	**-**	**-**	-	-
Greedy Best First Search	**No**	14	**12.36**	3373

2.1.1 Comparative Analysis

After considering all three (3) scenarios, among un-informed search strategies **Breadth First Search(BFS)** and **Uniform Cost Search** always find the optimal path length. If we need to be hard and fast to find the most optimal path length, I recommend using **BFS** over Uniform Cost Search as BFS is optimal and complete(AIMA-3.4.1). If we only consider time of execution and memory usage and not optimal path length: **Depth First Graph Search** outperforms among all and it's next best alternative would be Greedy Best First Search. This result doesn't come as a surprise after studying through the performance comparison below(AIMA-3.4.7) of these strategies.



3.4.7 Comparing uninformed search strategies

Figure 3.21 compares search strategies in terms of the four evaluation criteria set forth in Section 3.3.2. This comparison is for tree-search versions. For graph searches, the main differences are that depth-first search is complete for finite state spaces and that the space and time complexities are bounded by the size of the state space.

Criterion	Breadth-First	Uniform-Cost	Depth-First	Depth-Limited	Iterative Deepening	Bidirectional (if applicable)
Complete?	Yes^a	Yes^{a,b}	No	No	Yes^a	Yes^{a,d}
Time	$O(b^d)$	$O(b^{1+\lfloor C^*/\epsilon \rfloor})$	$O(b^m)$	$O(b^\ell)$	$O(b^d)$	$O(b^{d/2})$
Space	$O(b^d)$	$O(b^{1+\lfloor C^*/\epsilon \rfloor})$	$O(bm)$	$O(b\ell)$	$O(bd)$	$O(b^{d/2})$
Optimal?	Yes^c	Yes	No	No	Yes^c	Yes^{c,d}

Figure 3.21 Evaluation of tree-search strategies. b is the branching factor; d is the depth of the shallowest solution; m is the maximum depth of the search tree; l is the depth limit. Superscript caveats are as follows: ^{a} complete if b is finite; ^{b} complete if step costs $\geq \epsilon$ for positive ϵ; ^{c} optimal if step costs are all identical; ^{d} if both directions use breadth-first search.

2.2 Informed Search Strategies:

Problem 1 Results:

Strategy	**Optimal**	**Path Length**	**Time(sec)**	**Node Expansions**
A* Search with h1 heuristic	**Yes**	**6**	**0.03**	55
A* Search with Ignore Preconditions heuristic	**Yes**	**6**	**0.047**	41
A* Search with Level Sum heuristic	**Yes**	**6**	0.847	**11**

Problem 2 Results:

Strategy	**Optimal**	**Path Length**	**Time(sec)**	**Node Expansions**
A* Search with h1 heuristic	**Yes**	**9**	12.743	4852
A* Search with Ignore Preconditions heuristic	**Yes**	**9**	**4.397**	1450
A* Search with Level Sum heuristic	**Yes**	**9**	76.405	**86**

Problem 3 Results:

Strategy	**Optimal**	**Path Length**	**Time(sec)**	**Node Expansions**
A* Search with h1 heuristic	**Yes**	**12**	83.482	18236
A* Search with Ignore Preconditions heuristic	**Yes**	**12**	**31.378**	5040
A* Search with Level Sum heuristic	**Yes**	**12**	558.294	**403**

2.2.1 Comparative Analysis:

After considering all three (3) scenarios, all A* search strategies with heuristics yield and optimal solution. **A* Search with Ignore Preconditions heuristic** is the best when considering the execution time and **A* Search with Level Sum heuristic** is the best when considering the memory usage.

3. Final Analysis Recommendation

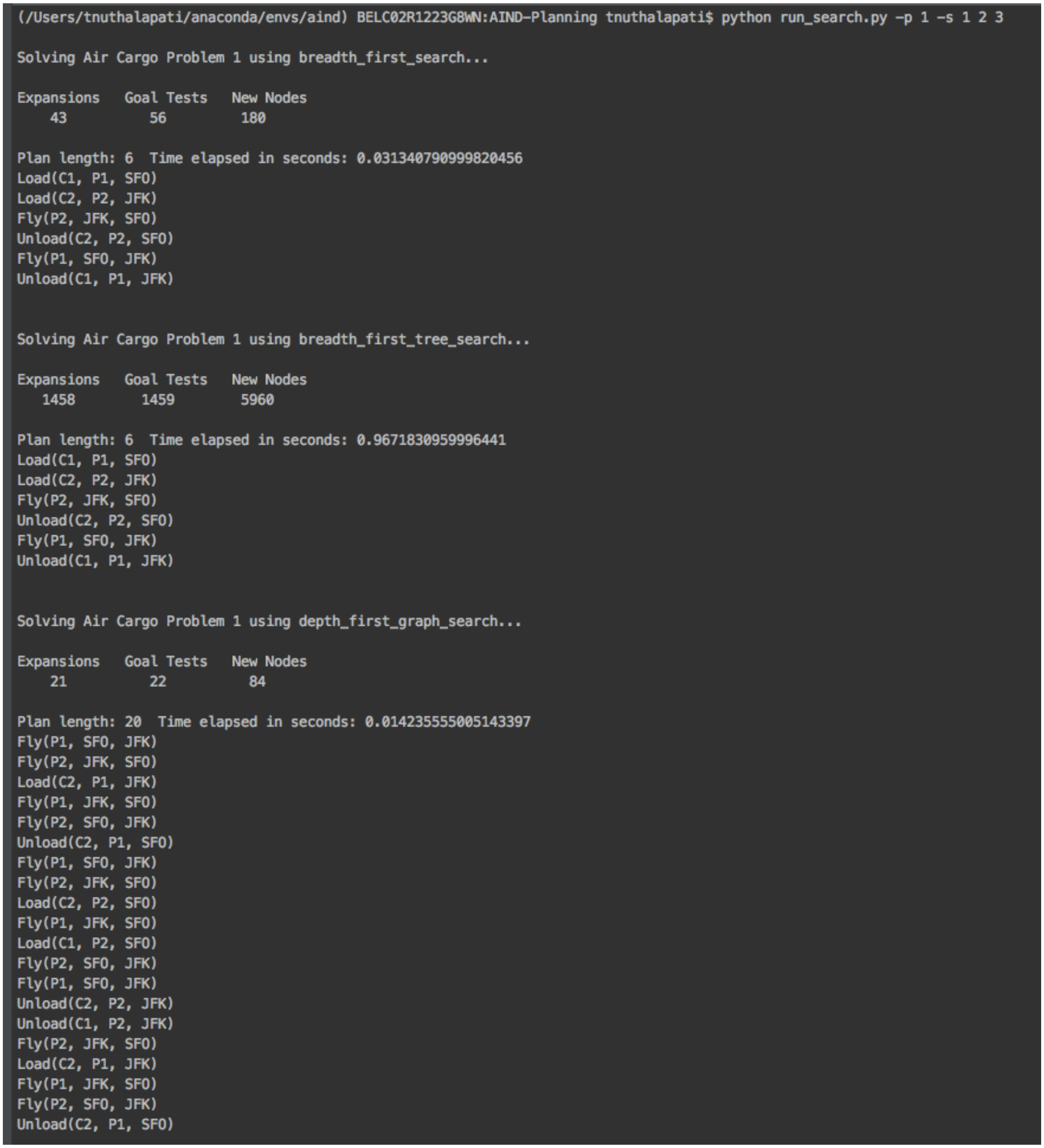
For Solution 3:

Strategy	Optimal	Path Length	Time(sec)	Node Expansions
BFS	**Yes**	**12**	117.22	14663
A* Search with Ignore Preconditions heuristic	**Yes**	**12**	**31.378**	5040

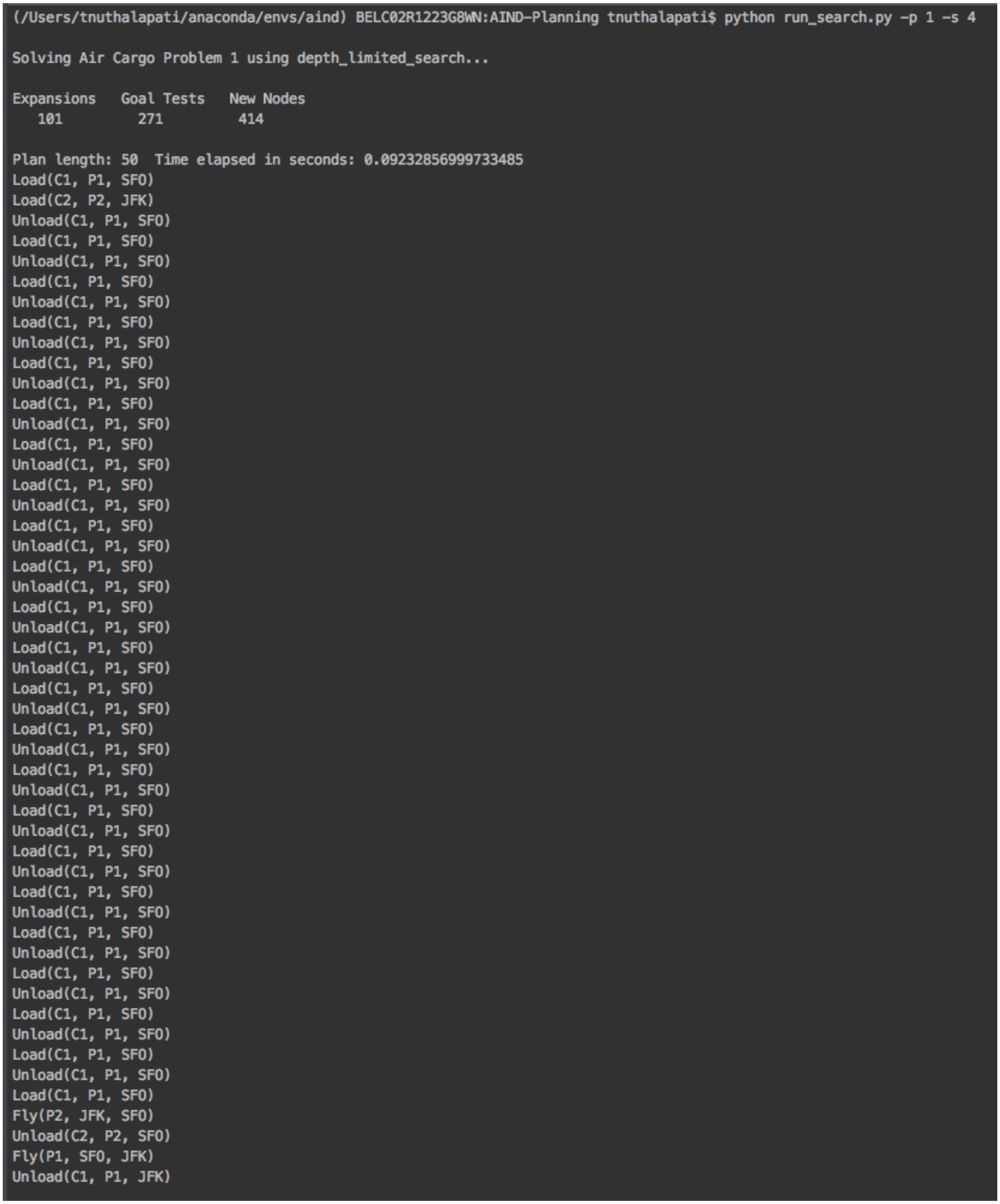
Whilst comparison among Informed & Un-Informed Search strategies recommendations of **Breadth First Search** and **A* Search with Ignore Preconditions heuristic**, it's the **A*** that outperforms BFS in terms of time and node expansions and is more optimal.

4. APPENDIX

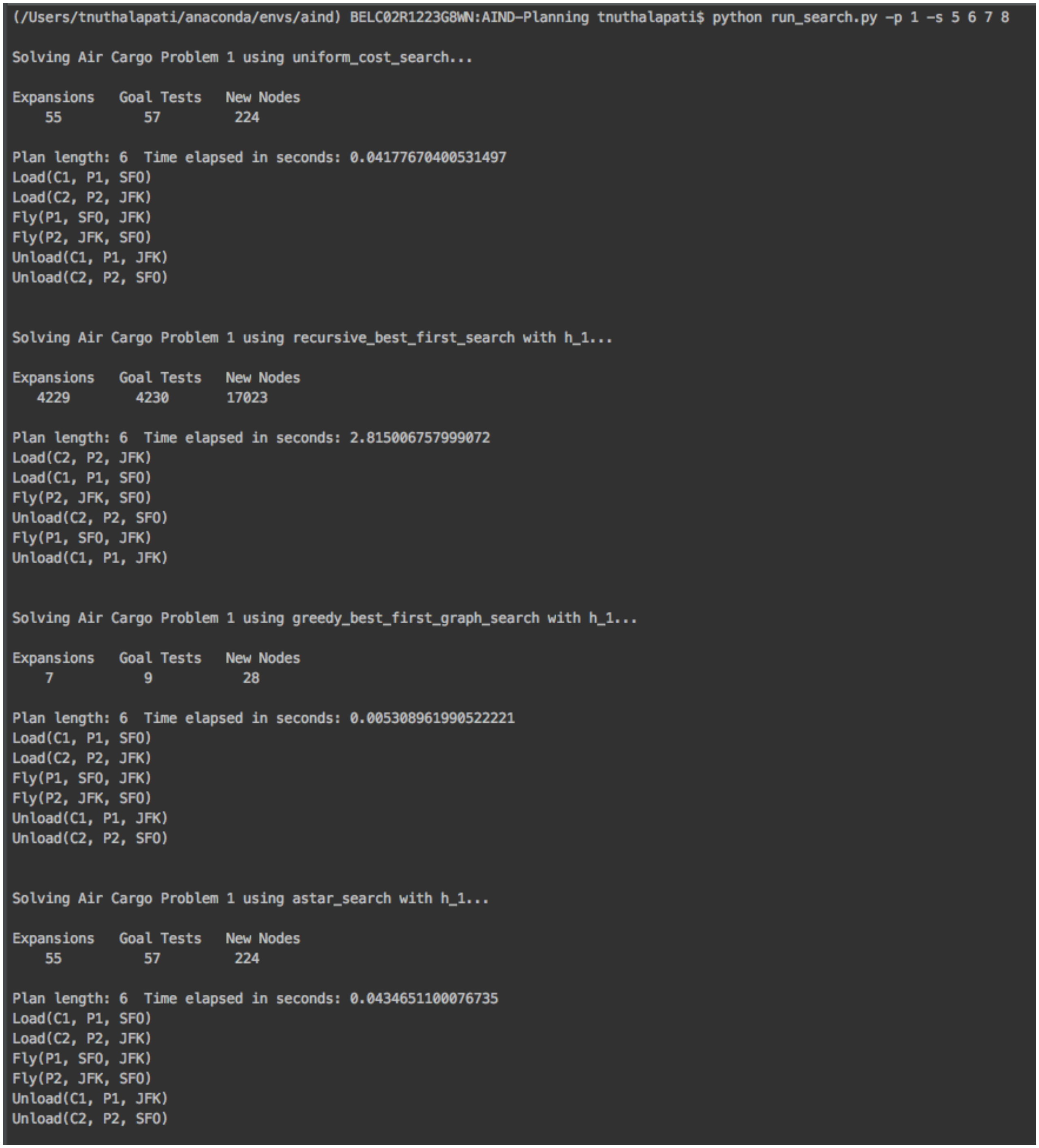
4.1 Results for Problem-1 using for all the heuristics 1-10



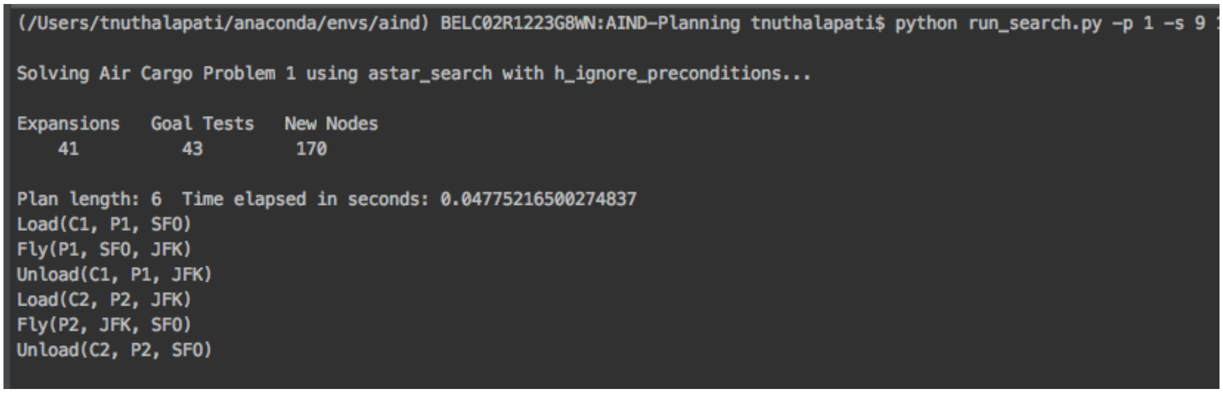
```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 1 -s 1 2 3

Solving Air Cargo Problem 1 using breadth_first_search...

Expansions   Goal Tests   New Nodes
    43          56          180

Plan length: 6  Time elapsed in seconds: 0.031340790999820456
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)


Solving Air Cargo Problem 1 using breadth_first_tree_search...

Expansions   Goal Tests   New Nodes
   1458        1459        5960

Plan length: 6  Time elapsed in seconds: 0.9671830959996441
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)


Solving Air Cargo Problem 1 using depth_first_graph_search...

Expansions   Goal Tests   New Nodes
    21          22           84

Plan length: 20  Time elapsed in seconds: 0.014235555005143397
Fly(P1, SFO, JFK)
Fly(P2, JFK, SFO)
Load(C2, P1, JFK)
Fly(P1, JFK, SFO)
Fly(P2, SFO, JFK)
Unload(C2, P1, SFO)
Fly(P1, SFO, JFK)
Fly(P2, JFK, SFO)
Load(C2, P2, SFO)
Fly(P1, JFK, SFO)
Load(C1, P2, SFO)
Fly(P2, SFO, JFK)
Fly(P1, SFO, JFK)
Unload(C2, P2, JFK)
Unload(C1, P2, JFK)
Fly(P2, JFK, SFO)
Load(C2, P1, JFK)
Fly(P1, JFK, SFO)
Fly(P2, SFO, JFK)
Unload(C2, P1, SFO)
```



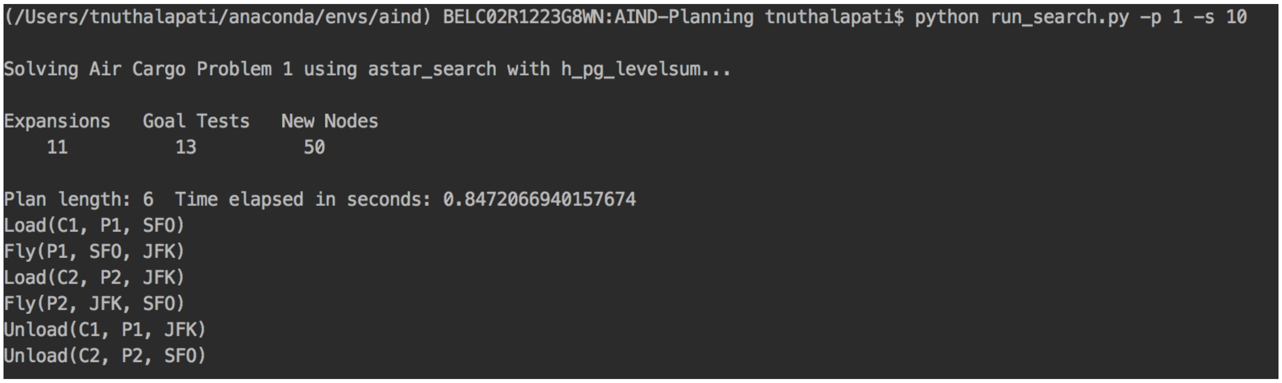
```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 1 -s 4

Solving Air Cargo Problem 1 using depth_limited_search...

Expansions   Goal Tests   New Nodes
   101          271         414

Plan length: 50  Time elapsed in seconds: 0.09232856999733485
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)
```



```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 1 -s 5 6 7 8

Solving Air Cargo Problem 1 using uniform_cost_search...

Expansions   Goal Tests   New Nodes
    55          57          224

Plan length: 6  Time elapsed in seconds: 0.04177670400531497
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P1, SFO, JFK)
Fly(P2, JFK, SFO)
Unload(C1, P1, JFK)
Unload(C2, P2, SFO)


Solving Air Cargo Problem 1 using recursive_best_first_search with h_1...

Expansions   Goal Tests   New Nodes
   4229        4230       17023

Plan length: 6  Time elapsed in seconds: 2.815006757999072
Load(C2, P2, JFK)
Load(C1, P1, SFO)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)


Solving Air Cargo Problem 1 using greedy_best_first_graph_search with h_1...

Expansions   Goal Tests   New Nodes
    7           9            28

Plan length: 6  Time elapsed in seconds: 0.005308961990522221
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P1, SFO, JFK)
Fly(P2, JFK, SFO)
Unload(C1, P1, JFK)
Unload(C2, P2, SFO)


Solving Air Cargo Problem 1 using astar_search with h_1...

Expansions   Goal Tests   New Nodes
    55          57          224

Plan length: 6  Time elapsed in seconds: 0.0434651100076735
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P1, SFO, JFK)
Fly(P2, JFK, SFO)
Unload(C1, P1, JFK)
Unload(C2, P2, SFO)
```

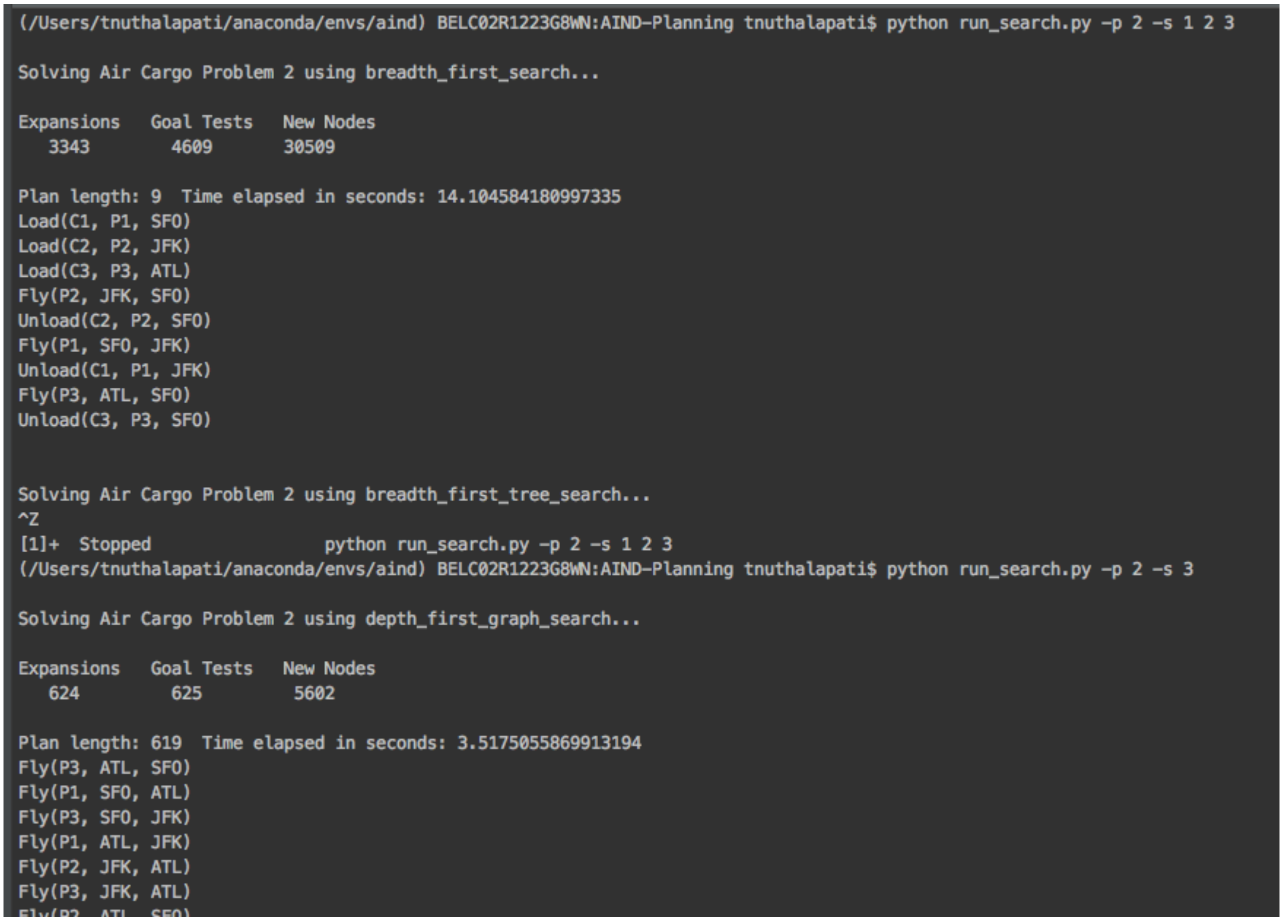


```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 1 -s 9 1

Solving Air Cargo Problem 1 using astar_search with h_ignore_preconditions...

Expansions   Goal Tests   New Nodes
    41          43         170

Plan length: 6  Time elapsed in seconds: 0.04775216500274837
Load(C1, P1, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
```

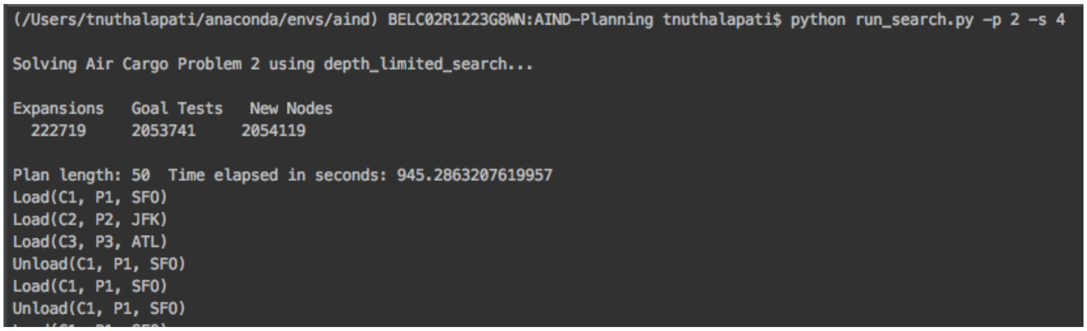


```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 1 -s 10

Solving Air Cargo Problem 1 using astar_search with h_pg_levelsum...

Expansions   Goal Tests   New Nodes
    11          13          50

Plan length: 6  Time elapsed in seconds: 0.8472066940157674
Load(C1, P1, SFO)
Fly(P1, SFO, JFK)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Unload(C1, P1, JFK)
Unload(C2, P2, SFO)
```

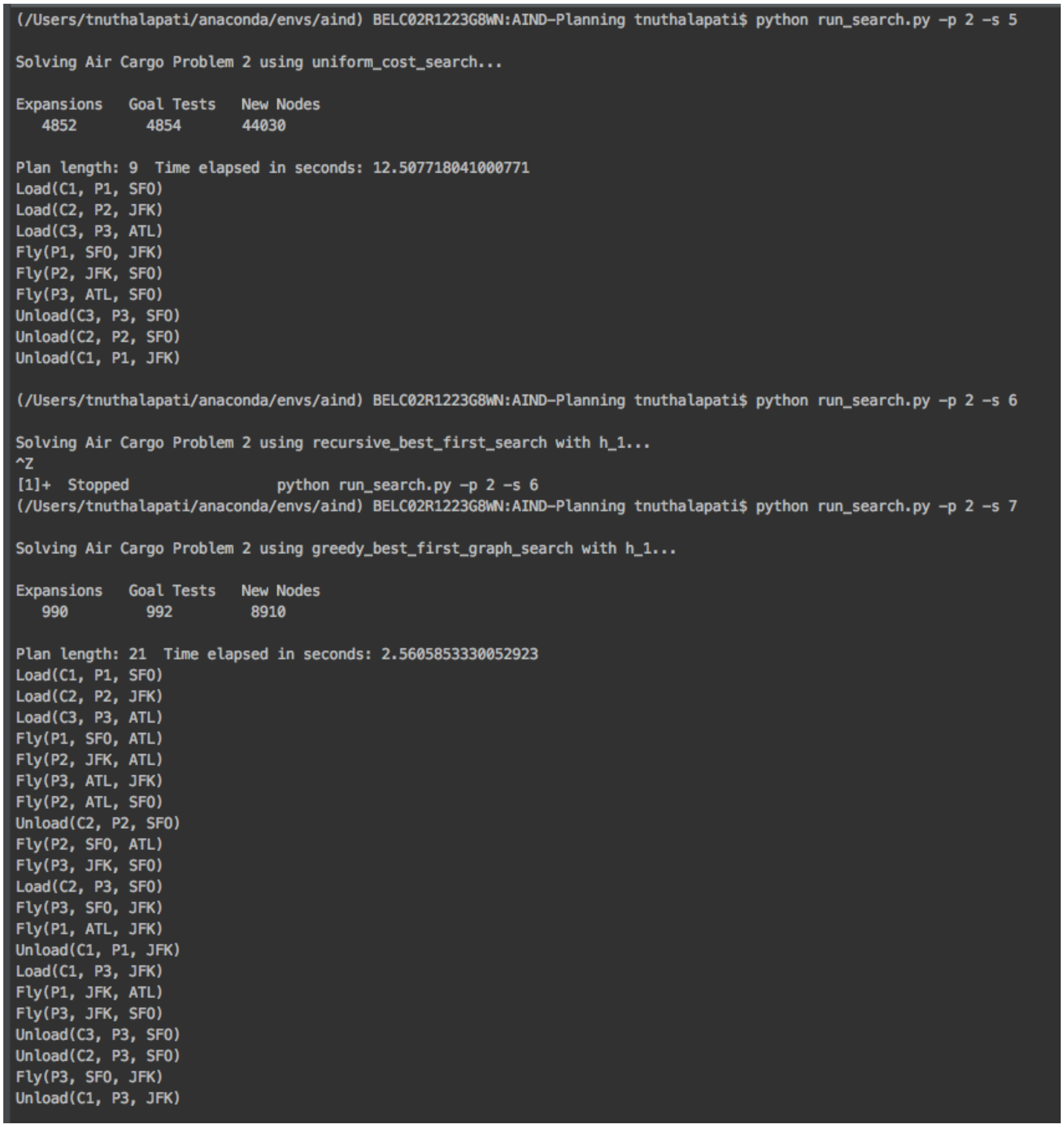
4.2 Results for Problem-2 using for all the heuristics 1-10



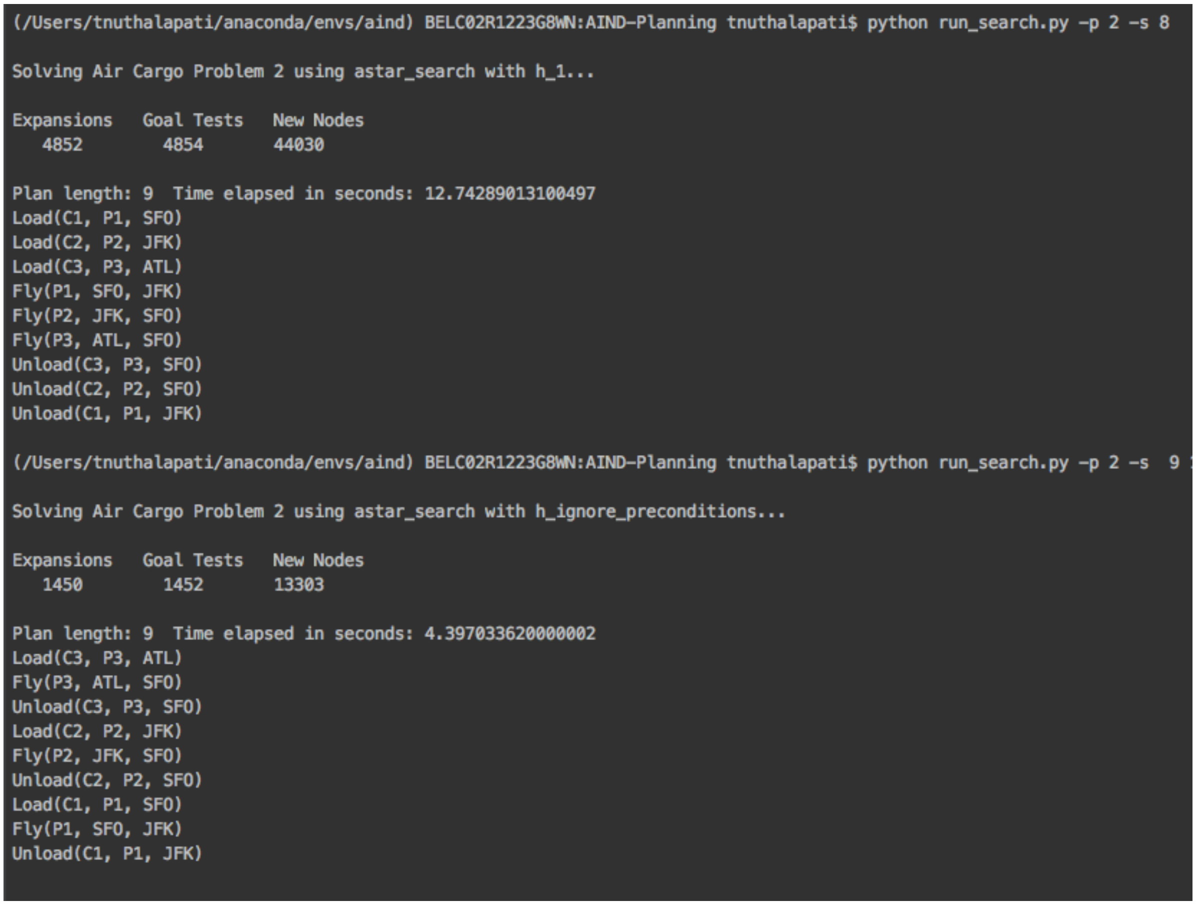
```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 2 -s 1 2 3

Solving Air Cargo Problem 2 using breadth_first_search...

Expansions   Goal Tests   New Nodes
   3343        4609       30509

Plan length: 9  Time elapsed in seconds: 14.104584180997335
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Load(C3, P3, ATL)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)
Fly(P3, ATL, SFO)
Unload(C3, P3, SFO)


Solving Air Cargo Problem 2 using breadth_first_tree_search...
^Z
[1]+  Stopped                 python run_search.py -p 2 -s 1 2 3
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 2 -s 3

Solving Air Cargo Problem 2 using depth_first_graph_search...

Expansions   Goal Tests   New Nodes
   624         625         5602

Plan length: 619  Time elapsed in seconds: 3.5175055869913194
Fly(P3, ATL, SFO)
Fly(P1, SFO, ATL)
Fly(P3, SFO, JFK)
Fly(P1, ATL, JFK)
Fly(P2, JFK, ATL)
Fly(P3, JFK, ATL)
Fly(P2, ATL, SFO)
```



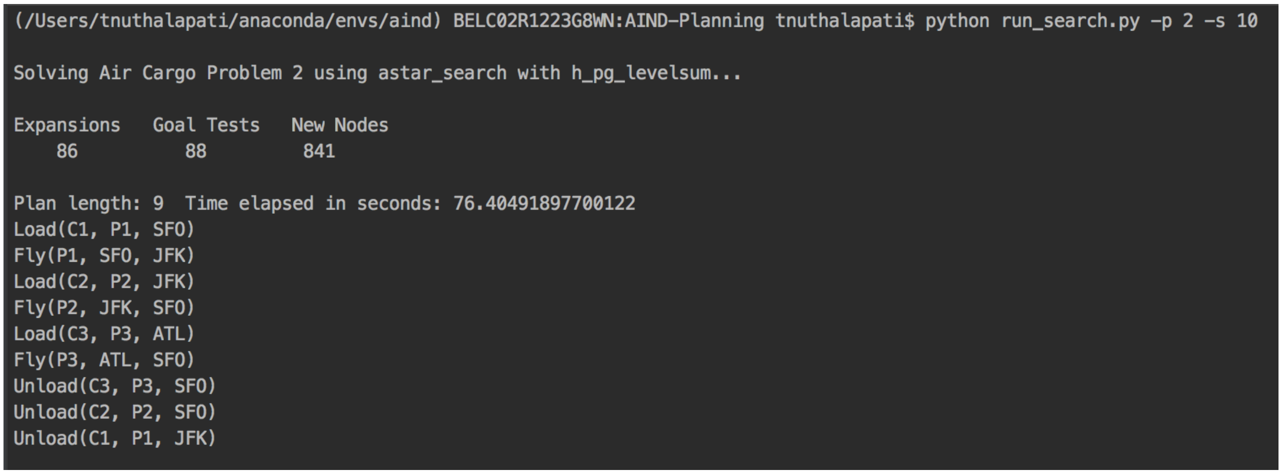
```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 2 -s 4

Solving Air Cargo Problem 2 using depth_limited_search...

Expansions   Goal Tests   New Nodes
  222719     2053741     2054119

Plan length: 50  Time elapsed in seconds: 945.2863207619957
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Load(C3, P3, ATL)
Unload(C1, P1, SFO)
Load(C1, P1, SFO)
Unload(C1, P1, SFO)
```



```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 2 -s 5

Solving Air Cargo Problem 2 using uniform_cost_search...

Expansions   Goal Tests   New Nodes
   4852        4854       44030

Plan length: 9  Time elapsed in seconds: 12.507718041000771
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Load(C3, P3, ATL)
Fly(P1, SFO, JFK)
Fly(P2, JFK, SFO)
Fly(P3, ATL, SFO)
Unload(C3, P3, SFO)
Unload(C2, P2, SFO)
Unload(C1, P1, JFK)

(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 2 -s 6

Solving Air Cargo Problem 2 using recursive_best_first_search with h_1...
^Z
[1]+  Stopped                 python run_search.py -p 2 -s 6
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 2 -s 7

Solving Air Cargo Problem 2 using greedy_best_first_graph_search with h_1...

Expansions   Goal Tests   New Nodes
   990         992         8910

Plan length: 21  Time elapsed in seconds: 2.5605853330052923
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Load(C3, P3, ATL)
Fly(P1, SFO, ATL)
Fly(P2, JFK, ATL)
Fly(P3, ATL, JFK)
Fly(P2, ATL, SFO)
Unload(C2, P2, SFO)
Fly(P2, SFO, ATL)
Fly(P3, JFK, SFO)
Load(C2, P3, SFO)
Fly(P3, SFO, JFK)
Fly(P1, ATL, JFK)
Unload(C1, P1, JFK)
Load(C1, P3, JFK)
Fly(P1, JFK, ATL)
Fly(P3, JFK, SFO)
Unload(C3, P3, SFO)
Unload(C2, P3, SFO)
Fly(P3, SFO, JFK)
Unload(C1, P3, JFK)
```

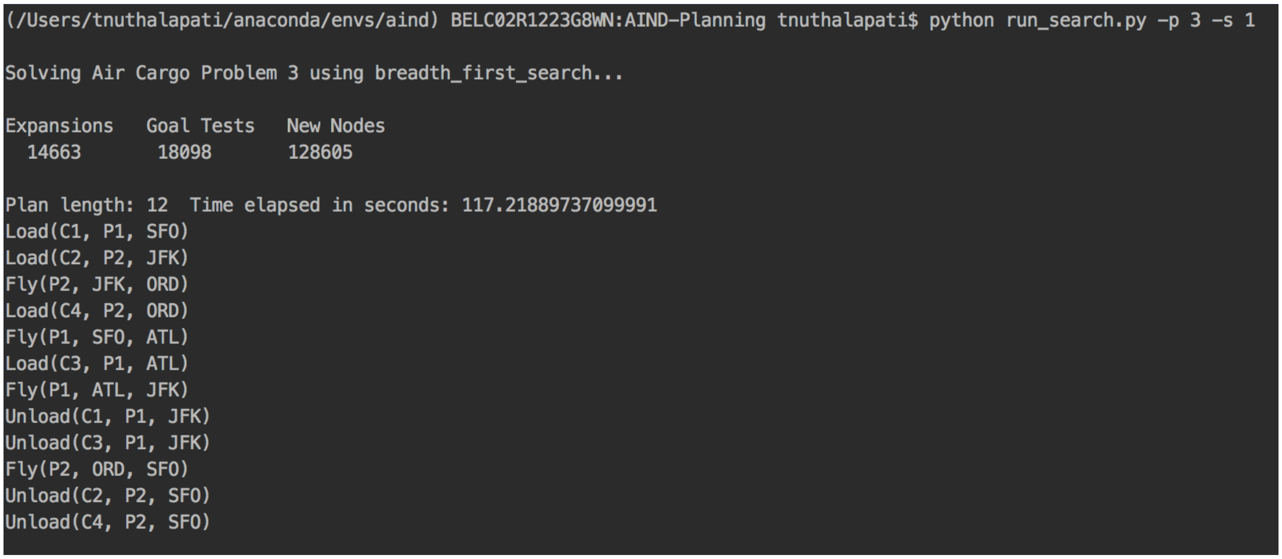


```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 2 -s 8

Solving Air Cargo Problem 2 using astar_search with h_1...

Expansions   Goal Tests   New Nodes
   4852        4854       44030

Plan length: 9  Time elapsed in seconds: 12.74289013100497
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Load(C3, P3, ATL)
Fly(P1, SFO, JFK)
Fly(P2, JFK, SFO)
Fly(P3, ATL, SFO)
Unload(C3, P3, SFO)
Unload(C2, P2, SFO)
Unload(C1, P1, JFK)

(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 2 -s  9

Solving Air Cargo Problem 2 using astar_search with h_ignore_preconditions...

Expansions   Goal Tests   New Nodes
   1450        1452       13303

Plan length: 9  Time elapsed in seconds: 4.397033620000002
Load(C3, P3, ATL)
Fly(P3, ATL, SFO)
Unload(C3, P3, SFO)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
Load(C1, P1, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)
```

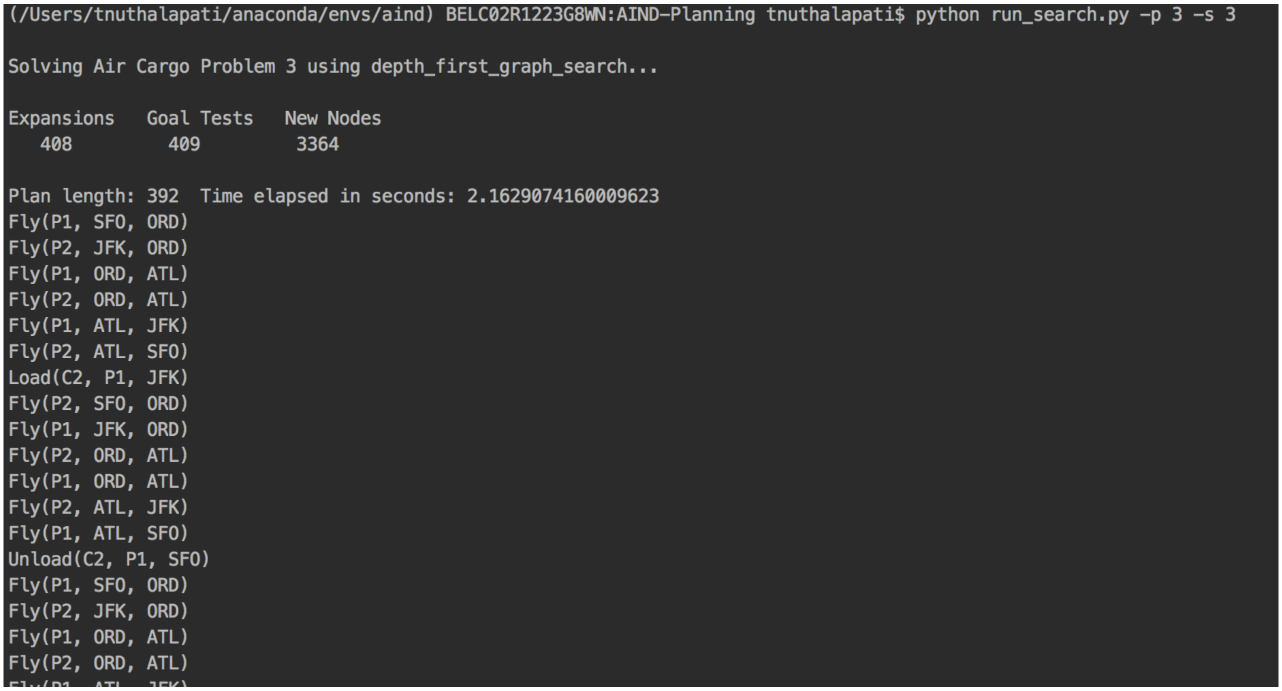


```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 2 -s 10

Solving Air Cargo Problem 2 using astar_search with h_pg_levelsum...

Expansions   Goal Tests   New Nodes
    86          88         841

Plan length: 9  Time elapsed in seconds: 76.40491897700122
Load(C1, P1, SFO)
Fly(P1, SFO, JFK)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Load(C3, P3, ATL)
Fly(P3, ATL, SFO)
Unload(C3, P3, SFO)
Unload(C2, P2, SFO)
Unload(C1, P1, JFK)
```

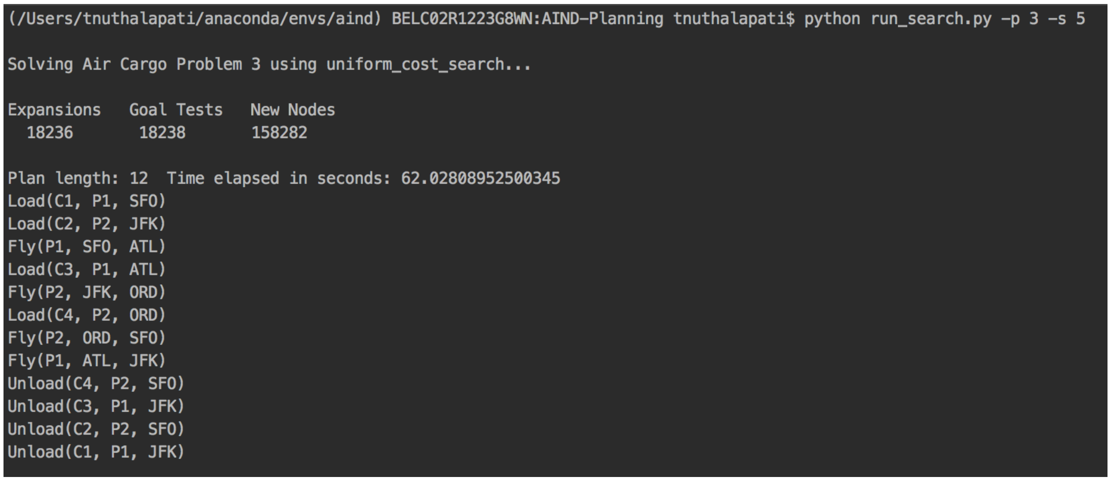
4.3 Results for Problem-3 using for all the heuristics 1-10



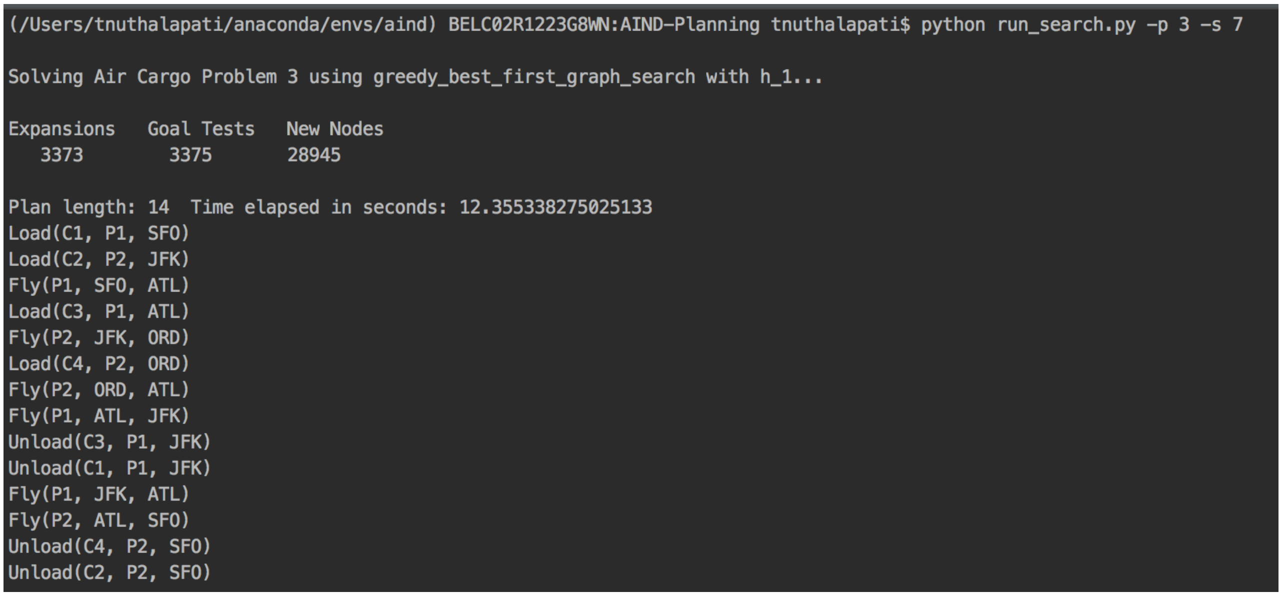
```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 3 -s 1

Solving Air Cargo Problem 3 using breadth_first_search...

Expansions   Goal Tests   New Nodes
  14663       18098       128605

Plan length: 12  Time elapsed in seconds: 117.21889737099991
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P2, JFK, ORD)
Load(C4, P2, ORD)
Fly(P1, SFO, ATL)
Load(C3, P1, ATL)
Fly(P1, ATL, JFK)
Unload(C1, P1, JFK)
Unload(C3, P1, JFK)
Fly(P2, ORD, SFO)
Unload(C2, P2, SFO)
Unload(C4, P2, SFO)
```



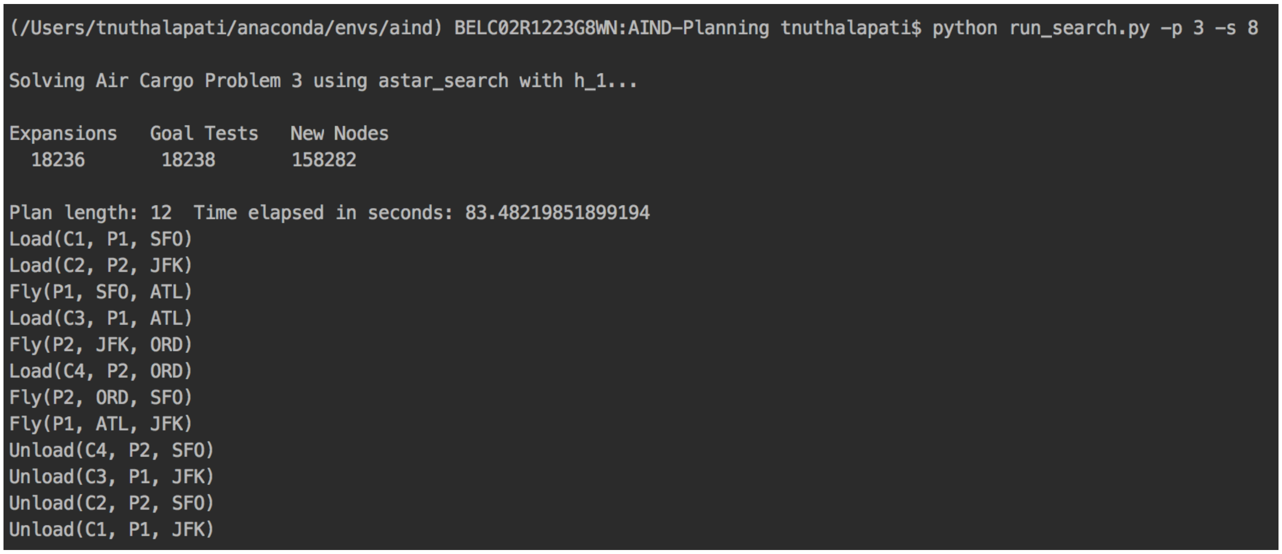
```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 3 -s 3

Solving Air Cargo Problem 3 using depth_first_graph_search...

Expansions   Goal Tests   New Nodes
   408         409         3364

Plan length: 392  Time elapsed in seconds: 2.1629074160009623
Fly(P1, SFO, ORD)
Fly(P2, JFK, ORD)
Fly(P1, ORD, ATL)
Fly(P2, ORD, ATL)
Fly(P1, ATL, JFK)
Fly(P2, ATL, SFO)
Load(C2, P1, JFK)
Fly(P2, SFO, ORD)
Fly(P1, JFK, ORD)
Fly(P2, ORD, ATL)
Fly(P1, ORD, ATL)
Fly(P2, ATL, JFK)
Fly(P1, ATL, SFO)
Unload(C2, P1, SFO)
Fly(P1, SFO, ORD)
Fly(P2, JFK, ORD)
Fly(P1, ORD, ATL)
Fly(P2, ORD, ATL)
Fly(P1, ATL, JFK)
```



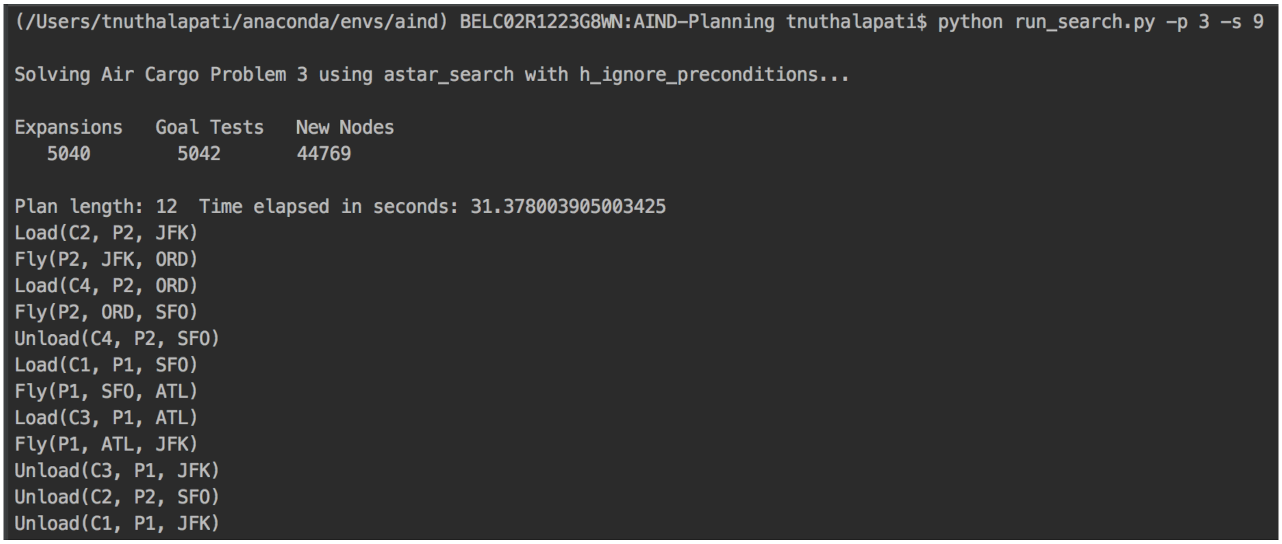
```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 3 -s 5

Solving Air Cargo Problem 3 using uniform_cost_search...

Expansions   Goal Tests   New Nodes
  18236       18238       158282

Plan length: 12  Time elapsed in seconds: 62.02808952500345
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P1, SFO, ATL)
Load(C3, P1, ATL)
Fly(P2, JFK, ORD)
Load(C4, P2, ORD)
Fly(P2, ORD, SFO)
Fly(P1, ATL, JFK)
Unload(C4, P2, SFO)
Unload(C3, P1, JFK)
Unload(C2, P2, SFO)
Unload(C1, P1, JFK)
```



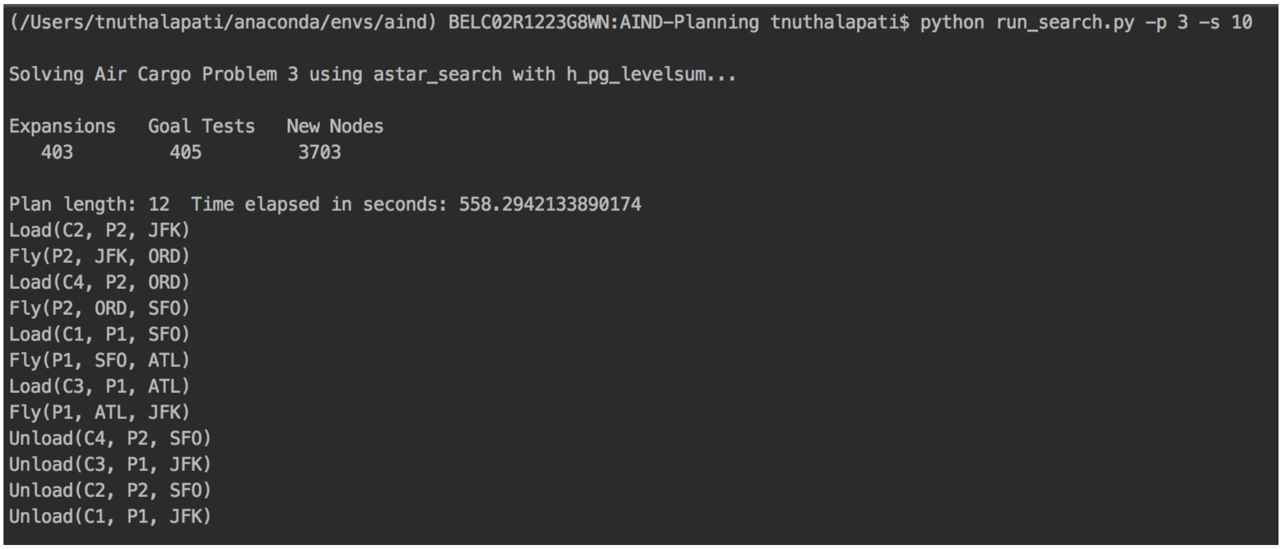
```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 3 -s 7

Solving Air Cargo Problem 3 using greedy_best_first_graph_search with h_1...

Expansions   Goal Tests   New Nodes
   3373        3375       28945

Plan length: 14  Time elapsed in seconds: 12.355338275025133
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P1, SFO, ATL)
Load(C3, P1, ATL)
Fly(P2, JFK, ORD)
Load(C4, P2, ORD)
Fly(P2, ORD, ATL)
Fly(P1, ATL, JFK)
Unload(C3, P1, JFK)
Unload(C1, P1, JFK)
Fly(P1, JFK, ATL)
Fly(P2, ATL, SFO)
Unload(C4, P2, SFO)
Unload(C2, P2, SFO)
```



```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 3 -s 8

Solving Air Cargo Problem 3 using astar_search with h_1...

Expansions   Goal Tests   New Nodes
  18236       18238       158282

Plan length: 12  Time elapsed in seconds: 83.48219851899194
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P1, SFO, ATL)
Load(C3, P1, ATL)
Fly(P2, JFK, ORD)
Load(C4, P2, ORD)
Fly(P2, ORD, SFO)
Fly(P1, ATL, JFK)
Unload(C4, P2, SFO)
Unload(C3, P1, JFK)
Unload(C2, P2, SFO)
Unload(C1, P1, JFK)
```



```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 3 -s 9

Solving Air Cargo Problem 3 using astar_search with h_ignore_preconditions...

Expansions   Goal Tests   New Nodes
   5040        5042       44769

Plan length: 12  Time elapsed in seconds: 31.378003905003425
Load(C2, P2, JFK)
Fly(P2, JFK, ORD)
Load(C4, P2, ORD)
Fly(P2, ORD, SFO)
Unload(C4, P2, SFO)
Load(C1, P1, SFO)
Fly(P1, SFO, ATL)
Load(C3, P1, ATL)
Fly(P1, ATL, JFK)
Unload(C3, P1, JFK)
Unload(C2, P2, SFO)
Unload(C1, P1, JFK)
```



```
(/Users/tnuthalapati/anaconda/envs/aind) BELC02R1223G8WN:AIND-Planning tnuthalapati$ python run_search.py -p 3 -s 10

Solving Air Cargo Problem 3 using astar_search with h_pg_levelsum...

Expansions   Goal Tests   New Nodes
   403         405         3703

Plan length: 12  Time elapsed in seconds: 558.2942133890174
Load(C2, P2, JFK)
Fly(P2, JFK, ORD)
Load(C4, P2, ORD)
Fly(P2, ORD, SFO)
Load(C1, P1, SFO)
Fly(P1, SFO, ATL)
Load(C3, P1, ATL)
Fly(P1, ATL, JFK)
Unload(C4, P2, SFO)
Unload(C3, P1, JFK)
Unload(C2, P2, SFO)
Unload(C1, P1, JFK)
```