ANALYSIS

Non Heuristic Searches-

Problem	Breadth_first_search	Depth_first_graph_search	Uniform_cost_search
Air Cargo Problem 1	Expansions-43 Goal Tests-56 New Nodes-180 Time- 0.030782620000536554 Optimal Path- Load(C1, P1, SFO) Load(C2, P2, JFK) Fly(P2, JFK, SFO) Unload(C2, P2, SFO) Fly(P1, SFO, JFK) Unload(C1, P1, JFK)	Expansions-21 Goal Tests-22 New Nodes-84 Time- 0.014805071994487662 Path length is 20- Hence not optimal	Expansions-55 Goal Tests-57 New Nodes-224 Time- 0.04118156401091255 Optimal Path- Load(C1, P1, SFO) Load(C2, P2, JFK) Fly(P1, SFO, JFK) Fly(P2, JFK, SFO) Unload(C1, P1, JFK) Unload(C2, P2, SFO)
Air Cargo Problem 2	Expansions-3346 Goal Tests-4612 New Nodes-30534 Time- 13.897042825992685 Optimal Path- Load(C1, P1, SFO) Load(C2, P2, JFK) Load(C3, P3, ATL) Fly(P1, SFO, JFK) Unload(C1, P1, JFK) Fly(P2, JFK, SFO) Unload(C2, P2, SFO) Fly(P3, ATL, SFO) Unload(C3, P3, SFO)	Expansions-107 Goal Tests-108 New Nodes-959 Time- 0.3423971490119584 Path length is 105- Hence not optimal	Expansions-4852 Goal Tests-4854 New Nodes-44030 Time- 45.0311206089973 Optimal Path- 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)
Air Cargo Problem 3	Expansions-3599 Goal Tests-6338 New Nodes-31930 Time- 20.77853937400505 Optimal Path-	Expansions-93 Goal Tests-94 New Nodes-783 Time- 0.280162385024596 Path length is 92-	Expansions-7368 Goal Tests-7370 New Nodes-66215 Time- 136.17521741401288 Optimal Path-

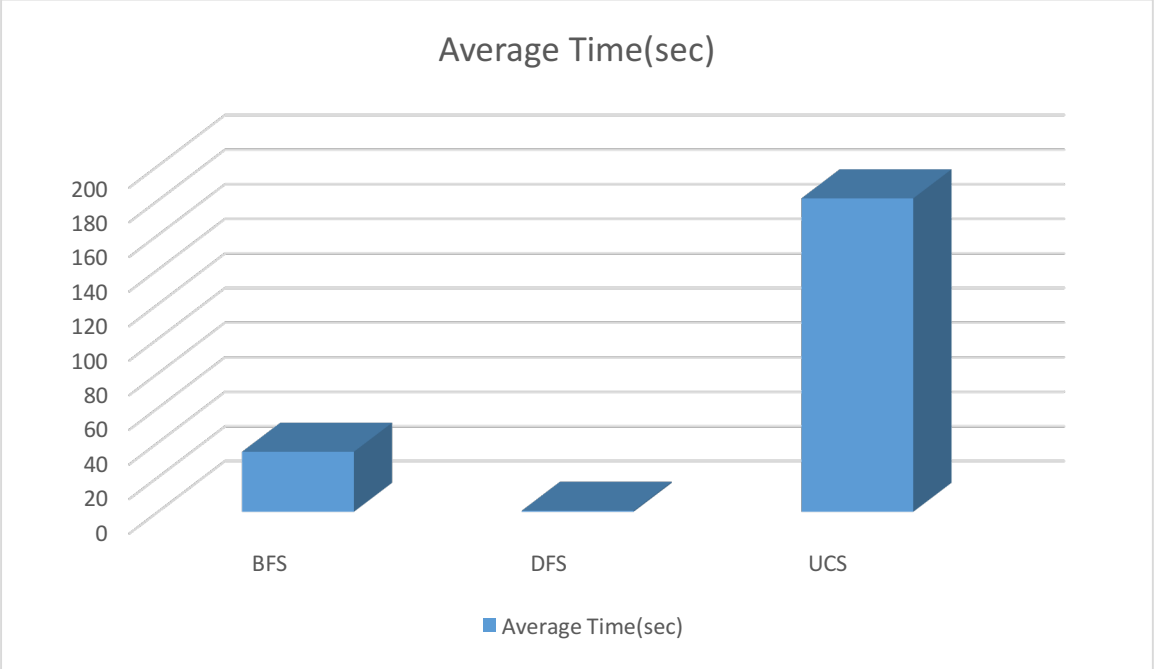
	Load(C1, P1, SFO) Load(C2, P2, JFK) Fly(P1, SFO, ATL) Load(C3, P1, ATL) Fly(P2, JFK, SFO) Unload(C2, P2, SFO) Fly(P1, ATL, JFK) Unload(C1, P1, JFK) Unload(C3, P1, JFK)	Hence not optimal	Load(C1, P1, SFO) Load(C2, P2, JFK) Fly(P1, SFO, ATL) Load(C3, P1, ATL) Fly(P2, JFK, SFO) Fly(P1, ATL, JFK) Unload(C3, P1, JFK) Unload(C2, P2, SFO) Unload(C1, P1, JFK)

- As it can be seen depth first search clearly takes the least amount of time to search but its searches are not optimal

- The path length generated by dfs is not optimal and is much more than bfs and uniform cost search

- BFS and uniform cost search generate similar paths to achieve the goals.

- But BFS takes less time than uniform cost search,hence BFS is the optimal searching technique amongst the three.



- BFS has an average searching time less than UCS in all the three cases.

- **Heuristic Searches-**

Problem	A*_h_ignore_preconditions	A*_h_pg_levelsum
Air Cargo Problem 1	Expansions-41 Goal Tests-43 New Nodes-170 Time- 0.03225685301003978 Optimal Path- Load(C1, P1, SFO) Fly(P1, SFO, JFK) Unload(C1, P1, JFK) Load(C2, P2, JFK) Fly(P2, JFK, SFO) Unload(C2, P2, SFO)	Expansions-11 Goal Tests-13 New Nodes-50 Time- 1.5453272760205436 Optimal Path- Load(C1, P1, SFO) Fly(P1, SFO, JFK) Load(C2, P2, JFK) Fly(P2, JFK, SFO) Unload(C1, P1, JFK) Unload(C2, P2, SFO)
Air Cargo Problem 2	Expansions-1506 Goal Tests-1508 New Nodes-13820 Time- 12.385466430016095 Optimal Path- 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)	Expansions- 86 Goal Tests- 88 New Nodes- 841 Time- 155.16828671601252 Optimal Path- 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)
Air Cargo Problem 3	Expansions- 870 Goal Tests- 872 New Nodes-7571 Time- 5.399042153003393 Optimal Path- Load(C1, P1, SFO) Fly(P1, SFO, ATL) Load(C3, P1, ATL)	Expansions-71 Goal Tests-73 New Nodes-625 Time- 195.00131706500542 Optimal Path- Load(C1, P1, SFO) Fly(P1, SFO, ATL) Load(C3, P1, ATL)

	Fly(P1, ATL, JFK) Unload(C3, P1, JFK) Load(C2, P2, JFK) Fly(P2, JFK, SFO) Unload(C2, P2, SFO) Unload(C1, P1, JFK)	Fly(P1, ATL, JFK) Load(C2, P2, JFK) Fly(P2, JFK, SFO) Unload(C3, P1, JFK) Unload(C2, P2, SFO) Unload(C1, P1, JFK)

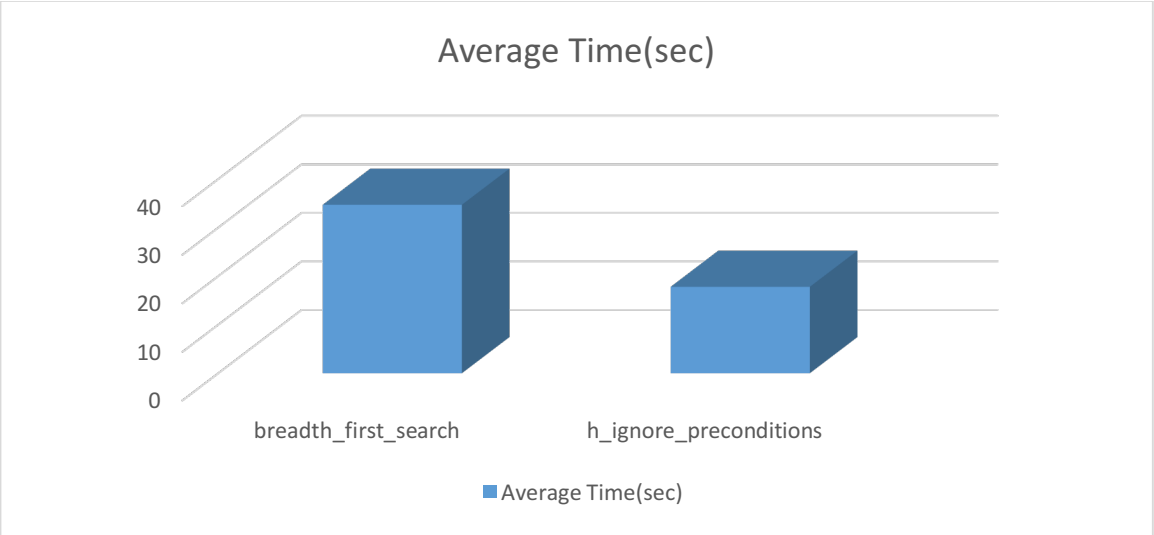
- The h_ignore_preconditions heuristic always takes less time than the h_levelsum for all the problems.

- But the h_ignore_precondtions heuristic expands more nodes as it does not constrain itself to any condtions and thus resulting into creation of more nodes.

- h_pg_levelsum does not expand many nodes as it is constrained to precontions and the effects.

- Both the heursitics generate optimal paths of same length and are very similar in the actions.

- But the h_ignore_preconditions must be considered optimal as it relaxes the problem and searches in lesser time.



- The heuristic method h_ignore_preconditions can be considered more optimal than the non heuristic bfs method as it takes a lot less time to search and also the expansions are significantly lesser.