Performance Summary

Optimal paths were found for all 3 problems, with slight variations on the order such as flying and unloading. Of all algorithms used depth first search was the quickest to find a solution, but in all cases created a suboptimal path. Where the number of steps does not matter one might prefer this method for the added speed increase (1 minute quicker than the next best solution in problem 3).

The A* with PG level sum method used fewer expansions and checking of all the other heuristics, but was by far the slowest to compute. The quickest optimal solution for each of the problems were as follows:

- Problem 1: Uniform Cost Search
- Problem 2: Uniform Cost Search
- Problem 3: A* Ignore Preconditions

With smaller problems the branching factor is much smaller which could explain why breadth-oriented algorithms such as uniform cost search outperforms heuristic guided methods. When comparing the A* methods ignoring the preconditions resulted in finding the optimal solution the quickest, checking fewer nodes than H_1 but more than calculating the level sums. If one were to scale the problem even further it would be preferred that this method should be used to obtain an optimal solution in the shortest amount of time.

Problem 1					
Algorithm	**Expansions**	**Goal Test**	**New Nodes**	**Time Taken(s)**	**Plan Length**
Breadth First Search	43	56	180	0.121255124	6
Depth First Search	21	22	84	0.074416912	20
Uniform Cost Search	55	57	224	0.169308516	6
A* with H_1	55	57	224	0.223031659	6
A* Ignore Preconditions	41	43	170	0.209135437	6
A* H_PG_levelsum	11	13	50	2.104969565	6
Problem 2					
Algorithm	**Expansions**	**Goal Test**	**New Nodes**	**Time Taken(s)**	**Plan Length**
Breadth First Search	3343	4609	30509	56.93450412	9
Depth First Search	624	625	5602	17.93033345	619
Uniform Cost Search	998	1000	8982	16.47030868	21
A* with H_1	4853	4855	44041	132.3660087	9
A* Ignore Preconditions	1450	1452	13303	43.83692329	9
A* H_PG_levelsum	86	88	841	259.8068469	9
Problem 3					
Algorithm	**Expansions**	**Goal Test**	**New Nodes**	**Time Taken(s)**	**Plan Length**
Breadth First Search	14663	18098	129631	327.3325522	12
Depth First Search	408	409	3364	10.44203788	392
Uniform Cost Search	18223	18225	159618	378.1099912	12
A* with H_1	18223	18225	159618	371.2105906	12
A* Ignore Preconditions	5040	5042	44944	79.10846495	12
A* H_PG_levelsum	325	327	3002	973.2551487	12

Albert Chung

Optimal Paths

These are the optimal paths for each problem found by Breadth First Search. Other methods created similar routes with slight reordering of load, fly and unload.

Problem 1
Load(C1, P1, SFO)
Load(C2, P2, JFK)
Fly(P2, JFK, SFO)
Unload(C2, P2, SFO)
Fly(P1, SFO, JFK)
Unload(C1, P1, JFK)

Problem 2
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
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)