**Optimal Action**

Below is a sample of the optimal action for Problem 1

Load(C1, P1, SFO)

Load(C2, P2, JFK)

Fly(P1, SFO, JFK)

Fly(P2, JFK, SFO)

Unload(C1, P1, JFK)

Unload(C2, P2, SFO)

Below is a sample of the optimal action for 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)

Below is a sample of the optimal action for 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)

**Uninformed Search**

**Problem 1**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Path | Time | Expansions |
| Breath First Search | 6 | .03 | 43 |
| Breath First Tree Search | 6 | .88 | 1458 |
| Depth First Graph Search | 12 | .007 | 12 |
| Depth First Limited Search | 50 | 0.09 | 101 |
| Uniform Cost Search | 6 | 0.033 | 55 |
| Recursive Best First Search | 6 | 2.7 | 4429 |
| Greedy Best First Graph Search | 6 | 0.004 | 7 |

**Problem 2**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Path | Time | Expansions |
| Breath First Search | 9 | 13.3 | 3343 |
| Depth First Graph Search | 466 | 2.3 | 476 |
| Uniform Cost Search | 9 | 11.8 | 4852 |
| Greedy Best First Graph Search | 9 | 2.2 | 990 |

**Problem 3**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Path | Time | Expansions |
| Breath First Search | 12 | 98.5 | 14663 |
| Depth First Graph Search | 1142 | 11.9 | 1511 |
| Uniform Cost Search | 12 | 49.3 | 18234 |
| Greedy Best First Graph Search | 22 | 16.1 | 5605 |

While BFS(Breath-First Search) and UCS(Uniform-Cost Search) both systematically yielded an optimal path for all three problems, UCS consistently took less time for it to complete, which was contrary to my expectation, as DFS consumes less memory and has a fewer number of expansions. Whether this behavior is specific to my hardware (Macbook Pro 15”) is subject to further investigation, UCS is the choice of algorithm if obtaining an optimal path is critical for now.

If obtaining an optimal path is not a requirement, Greedy Best First Graph Search is a good alternative as it has an optimal path for 2 out of 3 problems, and is much faster than both of the algorithms that lead to an optimal solution. In addition, it isn’t nearly as bad as depth first graph search, which seems to be very bad at finding an optimal path.

**Heuristic Search**

**Problem 1**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Path | Time | Expansions |
| A\* with h1 | 6 | 0.03 | 55 |
| A\* with h\_ignore | 6 | 0.03 | 41 |
| A\* with pagesum | 6 | 1.13 | 11 |

**Problem 2**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Path | Time | Expansions |
| A\* with h1 | 9 | 10.76 | 4852 |
| A\* with h\_ignore | 9 | 3.93 | 1450 |
| A\* with pagesum | 9 | 180.7 | 86 |

**Problem 3**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Path | Time | Expansions |
| A\* with h1 | 12 | 49.8 | 18234 |
| A\* with h\_ignore | 12 | 15.7 | 5040 |

A\* with pagesum for Problem 3 was not performed as the execution time was too long (10>m minute). All three yielded an optimal path, and A\* search with h\_ignore does so within the shortest amount of time and is much faster and expands a fewer number of nodes compared to uninformed search.