P1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Plan Methods | Expansions | Goal Tests | New Nodes | Plan Length | Time Elapsed |
| breadth\_first\_search | 43 | 56 | 180 | 6 | 0.0492 |
| breadth\_first\_tree\_search | 1458 | 1459 | 5960 | 6 | 1.46 |
| depth\_first\_graph\_search | 12 | 13 | 48 | 12 | 0.0133 |
| depth\_limited\_search | 101 | 271 | 414 | 50 | 0.148 |
| uniform\_cost\_search | 55 | 57 | 224 | 6 | 0.0547 |
| recursive\_best\_first\_search | 4229 | 4230 | 17029 | 6 | 4.098 |
| greedy\_best\_first\_graph\_search | 7 | 9 | 28 | 6 | 0.00813 |
| A\*Search(h\_1) | 55 | 57 | 224 | 6 | 0.0591 |
| A\*Search(h\_ignore\_preconditions) | 41 | 43 | 170 | 6 | 0.0605 |
| A\*Search(h\_pg\_levelsum) | 11 | 13 | 50 | 6 | 0.641 |

The optimal plan is:

Load(C1, P1, SFO)

Load(C2, P2, JFK)

Fly(P1, SFO, JFK)

Fly(P2, JFK, SFO)

Unload(C1, P1, JFK)

Unload(C2, P2, SFO)

Where the greedy\_best\_first\_graph\_search method is applied. From the table we can see this method performs best with minimal expansions, goal tests, new nodes, and shortest plan length and time elapsed.

P2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Plan Methods | Expansions | Goal Tests | New Nodes | Plan Length | Time Elapsed |
| breadth\_first\_search | 3346 | 4612 | 30534 | 9 | 11.6 |
| breadth\_first\_tree\_search | / | / | / | / | / |
| depth\_first\_graph\_search | 1124 | 1125 | 10017 | 1085 | 11.1 |
| depth\_limited\_search | / | / | / | / | / |
| uniform\_cost\_search | 4853 | 4855 | 44041 | 9 | 15.9 |
| recursive\_best\_first\_search | / | / | / | / | / |
| greedy\_best\_first\_graph\_search | 998 | 1000 | 8982 | 21 | 3.39 |
| A\*Search(h\_1) | 4853 | 4855 | 44041 | 9 | 17.9 |
| A\*Search(h\_ignore\_preconditions) | 1450 | 1452 | 13303 | 9 | 6.22 |
| A\*Search(h\_pg\_levelsum) | 86 | 88 | 841 | 9 | 59.6 |

The optimal plan is:

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)

Where A\*Search(h\_pg\_levelsum) method is applied. From the table we can see this method has minimal expansions, goal tests, new nodes, and shortest plan length, however, longest time.

P3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Plan Methods | Expansions | Goal Tests | New Nodes | Plan Length | Time Elapsed |
| breadth\_first\_search | 14120 | 17673 | 124926 | 12 | 57.3 |
| breadth\_first\_tree\_search | / | / | / | / | / |
| depth\_first\_graph\_search | 677 | 678 | 5608 | 660 | 4.98 |
| depth\_limited\_search | / | / | / | / | / |
| uniform\_cost\_search | 18223 | 18225 | 159618 | 12 | 75.3 |
| recursive\_best\_first\_search | / | / | / | / | / |
| greedy\_best\_first\_graph\_search | 5578 | 5580 | 49150 | 22 | 22.1 |
| A\*Search(h\_1) | 18223 | 18225 | 159618 | 12 | 74.9 |
| A\*Search(h\_ignore\_preconditions) | 5040 | 5042 | 44944 | 12 | 25.1 |
| A\*Search(h\_pg\_levelsum) | 325 | 327 | 3002 | 12 | 290 |

The optimal plan is:

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)