Air Cargo Problem Heuristic Analysis

Problem 1:

All the search algorithms were able to find the solution in this case. recursive\_best\_first\_search h\_1 alone took more than 2 seconds to find the solution. However, all the search algorithms except depth\_first\_graph\_search and depth\_limited\_search were able to find the solution in under one second and a plan length of 6.

The best search algorithm for this problem was ***greedy\_best\_first\_graph\_search h\_1.*** It solved the problem in 0.004 seconds with 7 expansions.

Problem 2:



In case of problem 2, breadth\_first\_tree\_search, depth\_limited\_search and recursive\_best\_first\_search h\_1 were not able to find the solution in under 10 minutes. Apart from depth\_first\_graph\_search, greedy\_best\_first\_graph\_search h\_1 and above mentioned three search algorithms all the other search algorithms were able to find the solution with a plan length of 9.

The best algorithm for this problem was A\* h\_ignore\_preconditions which got to the solution in 3.002 seconds with 1450 expansions. breadth\_first\_search got to the solution in 10.92 seconds with 3343 expansions. uniform\_cost\_search and A\* h\_1 were the next best after the above two search algorithms.

Problem 3:



In case of problem 3, similar to problem 2 breadth\_first\_tree\_search, depth\_limited\_search and recursive\_best\_first\_search h\_1 were not able to find the solution in under 10 minutes. Apart from depth\_first\_graph\_search, greedy\_best\_first\_graph\_search h\_1 and above mentioned three search algorithms all the other search algorithms were able to find the solution with a plan length of 12.

The best algorithm for this problem was A\* h\_ignore\_preconditions which got to the solution in 12.06 seconds with 5040 expansions. breadth\_first\_search got to the solution in 79.51 seconds with 14663 expansions. Similar to problem 2 uniform\_cost\_search and A\* h\_1 were the next best after the above two search algorithms.

Observations:

* In case of small problems all the search algorithms were able to find the solution but as the problem gets larger breadth\_first\_tree\_search, depth\_limited\_search and recursive\_best\_first\_search h\_1 were not able to find the optimal solution in limited time.
* breadth\_first\_search performed well but A\* h\_ignore\_preconditions beat the algorithm by finding the optimal plan with least expansions. However, breadth\_first\_search is the best non-heuristic search algorithm.
* The heuristic approach got to the optimal plan with less number of expansions compared to non-heuristic approach.
* The above tests were conducted on Macbook Pro [i7, 2.8 GHz and 16 GB RAM]