

# Analysis Report

## Data Cleaning

After implementing all the required algorithms for the assignment, I ran all the combinations required and a combination with all problems and search algorithms. I output all the logs to .txt files with the following code. I was installed pypy3 to help speed up the calculations. I let it run for about 3 hours.

-----

# You should run all of the search algorithms on the first two problems and record the following information for each combination.

```
pypy3 run_search.py -p 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 -s 1 2 3 4 5 6 7 8 9 10 11 1 2 3 4 5 6 7 8 9 10 11 > log_output_p1.txt
```

# You must run at least one uninformed search, two heuristics with greedy best first, and two heuristics with A\* on problems 3 and 4.

```
pypy3 run_search.py -p 3 3 3 3 3 -s 2 4 6 8 10 > log_output_p2.txt
```

# Output every search on every problem

```
pypy3 run_search.py -p 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 -s 1 2 3 4 5 6 7 8 9 10 11 1 2 3 4 5 6 7 8 9 10 11 1 2 3 4 5 6 7 8 9 10 11 1 2 3 4 5 6 7 8 9 10 11 > log_output_all.txt
```

## Reading raw/un-cleaned data from file

I placed the 3 files ('log\_output\_p1.txt', 'log\_output\_p2.txt', 'log\_output\_all.txt') in a sub-directory called 'data'. I used 'log\_output\_all.txt' but resaved it as 'log\_output\_all\_a.txt' to have the string encoding work with Python.

```
In [1]: file_lines_list = []
        with open('data/log_output_all_a.txt', 'r') as fp:
            for line in fp:
                file_lines_list.append( line )
```

```
In [2]: # Check the data captured
        file_lines_list[:10]
```

```
Out[2]: ['\n',
        'Solving Air Cargo Problem 1 using breadth_first_search...\n',
        '\n',
        '# Actions    Expansions    Goal Tests    New Nodes\n',
        '    20          43          56          178    \n',
        '\n',
        'Plan length: 6   Time elapsed in seconds: 0.02428653135087065\n',
        'Load(C1, P1, SFO)\n',
        'Load(C2, P2, JFK)\n',
        'Fly(P2, JFK, SFO)\n']
```

```
In [3]: import re
```

```

In [4]: # Filter just for information
dict_filtered = {}
for i, line in enumerate(file_lines_list):
    if not line.startswith('Solving'):
        continue

    tokens_a = line.replace('Solving ', '').replace('...\n', '').split('using')
    sub_line_a = file_lines_list[i + 3]
    sub_line_b = file_lines_list[i + 5]
    tokens_b = sub_line_a.strip().split()
    #tokens_c = [int(s) for s in sub_line_b.split() if s.isdigit()]
    tokens_c = re.findall(r"[-+]?[d*\.|d+]", sub_line_b)

    #      # DEBUG
    #      #
    #      print("DEBUG - sub_line_b: {}".format(sub_line_b))
    #      print("DEBUG - tokens_c: {}".format(tokens_c))
    #      break

    problem = str( tokens_a[0] ).strip()
    search = str( tokens_a[1] ).strip()
    actions = int( tokens_b[0] )
    expansions = int( tokens_b[1] )
    goal_tests = int( tokens_b[2] )
    new_nodes = int( tokens_b[3] )
    plan_length = int( tokens_c[0] )
    time_in_sec = float( tokens_c[1] )

    # Populate dictionary
    if problem not in dict_filtered.keys():
        dict_filtered[problem] = {}

    dict_filtered[problem][search] = {
        'actions'      : actions,
        'expansions'    : expansions,
        'goal_tests'    : goal_tests,
        'new_nodes'     : new_nodes,
        'plan_length'   : plan_length,
        'time_in_sec'   : time_in_sec }

```

```
In [5]: # Format data structure to be ingested by Pandas DataFrame.  
# This format will be a Python Dictionary formatted to a Pandas Row Oriented DataFrame  
list_of_dicts_row_oriented = []  
for i, problem in enumerate(dict_filtered.keys()):  
  
    for j, search in enumerate(dict_filtered[problem].keys()):  
  
        dict_row = {}  
        dict_row['problem'] = problem  
        dict_row['search'] = search  
  
        for k, info in enumerate(dict_filtered[problem][search].keys()):  
  
            dict_row[info] = dict_filtered[problem][search][info]  
  
        list_of_dicts_row_oriented.append(dict_row)
```

```
In [6]: # Check what the data structure looks like.  
dict_filtered
```

```
Out[6]: {'Air Cargo Problem 1': {'astar_search with h_pg_levelsum': {'actions': 20,
    'expansions': 28,
    'goal_tests': 30,
    'new_nodes': 122,
    'plan_length': 6,
    'time_in_sec': 0.21640066376721034},
    'astar_search with h_pg_maxlevel': {'actions': 20,
    'expansions': 43,
    'goal_tests': 45,
    'new_nodes': 180,
    'plan_length': 6,
    'time_in_sec': 0.12295208562681625},
    'astar_search with h_pg_setlevel': {'actions': 20,
    'expansions': 33,
    'goal_tests': 35,
    'new_nodes': 138,
    'plan_length': 6,
    'time_in_sec': 0.29609892190720943},
    'astar_search with h_unmet_goals': {'actions': 20,
    'expansions': 50,
    'goal_tests': 52,
    'new_nodes': 206,
    'plan_length': 6,
    'time_in_sec': 0.015270101490373222},
    'breadth_first_search': {'actions': 20,
    'expansions': 43,
    'goal_tests': 56,
    'new_nodes': 178,
    'plan_length': 6,
    'time_in_sec': 0.02428653135087065},
    'depth_first_graph_search': {'actions': 20,
    'expansions': 21,
    'goal_tests': 22,
    'new_nodes': 84,
    'plan_length': 20,
    'time_in_sec': 0.006418696626931376},
    'greedy_best_first_graph_search with h_pg_levelsum': {'actions': 20,
    'expansions': 6,
    'goal_tests': 8,
    'new_nodes': 28,
    'plan_length': 6,
    'time_in_sec': 0.3834521536819968},
    'greedy_best_first_graph_search with h_pg_maxlevel': {'actions': 20,
```

```
'expansions': 6,  
'goal_tests': 8,  
'new_nodes': 24,  
'plan_length': 6,  
'time_in_sec': 0.08394447825654622},  
'greedy_best_first_graph_search with h_pg_setlevel': {'actions': 20,  
'expansions': 6,  
'goal_tests': 8,  
'new_nodes': 28,  
'plan_length': 6,  
'time_in_sec': 0.41804252144090803},  
'greedy_best_first_graph_search with h_unmet_goals': {'actions': 20,  
'expansions': 7,  
'goal_tests': 9,  
'new_nodes': 29,  
'plan_length': 6,  
'time_in_sec': 0.0026820088196692904},  
'uniform_cost_search': {'actions': 20,  
'expansions': 60,  
'goal_tests': 62,  
'new_nodes': 240,  
'plan_length': 6,  
'time_in_sec': 0.01970920338766699}},  
'Air Cargo Problem 2': {'astar_search with h_pg_levelsum': {'actions': 72,  
'expansions': 357,  
'goal_tests': 359,  
'new_nodes': 3426,  
'plan_length': 9,  
'time_in_sec': 16.25889883191922},  
'astar_search with h_pg_maxlevel': {'actions': 72,  
'expansions': 2887,  
'goal_tests': 2889,  
'new_nodes': 26594,  
'plan_length': 9,  
'time_in_sec': 71.23779185270179},  
'astar_search with h_pg_setlevel': {'actions': 72,  
'expansions': 1037,  
'goal_tests': 1039,  
'new_nodes': 9605,  
'plan_length': 9,  
'time_in_sec': 134.70260450348985},  
'astar_search with h_unmet_goals': {'actions': 72,  
'expansions': 2467,
```

```
'goal_tests': 2469,  
'new_nodes': 22522,  
'plan_length': 9,  
'time_in_sec': 0.7492702580525501},  
'breadth_first_search': {'actions': 72,  
'expansions': 3343,  
'goal_tests': 4609,  
'new_nodes': 30503,  
'plan_length': 9,  
'time_in_sec': 0.3474298061943615},  
'depth_first_graph_search': {'actions': 72,  
'expansions': 624,  
'goal_tests': 625,  
'new_nodes': 5602,  
'plan_length': 619,  
'time_in_sec': 0.5629788986626627},  
'greedy_best_first_graph_search with h_pg_levelsum': {'actions': 72,  
'expansions': 9,  
'goal_tests': 11,  
'new_nodes': 86,  
'plan_length': 9,  
'time_in_sec': 0.6934766464945943},  
'greedy_best_first_graph_search with h_pg_maxlevel': {'actions': 72,  
'expansions': 27,  
'goal_tests': 29,  
'new_nodes': 249,  
'plan_length': 9,  
'time_in_sec': 0.6004847080362232},  
'greedy_best_first_graph_search with h_pg_setlevel': {'actions': 72,  
'expansions': 9,  
'goal_tests': 11,  
'new_nodes': 84,  
'plan_length': 9,  
'time_in_sec': 1.0814222051712061},  
'greedy_best_first_graph_search with h_unmet_goals': {'actions': 72,  
'expansions': 17,  
'goal_tests': 19,  
'new_nodes': 170,  
'plan_length': 9,  
'time_in_sec': 0.05454464031955597},  
'uniform_cost_search': {'actions': 72,  
'expansions': 5154,  
'goal_tests': 5156,
```



```
'new_nodes': 46618,  
'plan_length': 9,  
'time_in_sec': 0.7676067214540252}},  
'Air Cargo Problem 3': {'astar_search with h_pg_levelsum': {'actions': 88,  
'expansions': 369,  
'goal_tests': 371,  
'new_nodes': 3403,  
'plan_length': 12,  
'time_in_sec': 43.686991657774826},  
'astar_search with h_pg_maxlevel': {'actions': 88,  
'expansions': 9580,  
'goal_tests': 9582,  
'new_nodes': 86312,  
'plan_length': 12,  
'time_in_sec': 240.99240215040265},  
'astar_search with h_pg_setlevel': {'actions': 88,  
'expansions': 3423,  
'goal_tests': 3425,  
'new_nodes': 31596,  
'plan_length': 12,  
'time_in_sec': 317.41891877596595},  
'astar_search with h_unmet_goals': {'actions': 88,  
'expansions': 7388,  
'goal_tests': 7390,  
'new_nodes': 65711,  
'plan_length': 12,  
'time_in_sec': 1.6978325306583315},  
'breadth_first_search': {'actions': 88,  
'expansions': 14663,  
'goal_tests': 18098,  
'new_nodes': 129625,  
'plan_length': 12,  
'time_in_sec': 1.3805018029333667},  
'depth_first_graph_search': {'actions': 88,  
'expansions': 408,  
'goal_tests': 409,  
'new_nodes': 3364,  
'plan_length': 392,  
'time_in_sec': 0.36805228894607467},  
'greedy_best_first_graph_search with h_pg_levelsum': {'actions': 88,  
'expansions': 14,  
'goal_tests': 16,  
'new_nodes': 126,
```

```
'plan_length': 14,  
'time_in_sec': 2.6483832134316003},  
'greedy_best_first_graph_search with h_pg_maxlevel': {'actions': 88,  
'expansions': 21,  
'goal_tests': 23,  
'new_nodes': 195,  
'plan_length': 13,  
'time_in_sec': 1.2394026095106483},  
'greedy_best_first_graph_search with h_pg_setlevel': {'actions': 88,  
'expansions': 35,  
'goal_tests': 37,  
'new_nodes': 345,  
'plan_length': 17,  
'time_in_sec': 9.081586877550762},  
'greedy_best_first_graph_search with h_unmet_goals': {'actions': 88,  
'expansions': 25,  
'goal_tests': 27,  
'new_nodes': 230,  
'plan_length': 15,  
'time_in_sec': 0.019154536036438685},  
'uniform_cost_search': {'actions': 88,  
'expansions': 18510,  
'goal_tests': 18512,  
'new_nodes': 161936,  
'plan_length': 12,  
'time_in_sec': 2.316739035534681}},  
'Air Cargo Problem 4': {'astar_search with h_pg_levelsum': {'actions': 104,  
'expansions': 1208,  
'goal_tests': 1210,  
'new_nodes': 12210,  
'plan_length': 15,  
'time_in_sec': 195.75180205181073},  
'astar_search with h_pg_maxlevel': {'actions': 104,  
'expansions': 62077,  
'goal_tests': 62079,  
'new_nodes': 599376,  
'plan_length': 14,  
'time_in_sec': 2261.118180112795},  
'astar_search with h_pg_setlevel': {'actions': 104,  
'expansions': 22606,  
'goal_tests': 22608,  
'new_nodes': 224229,  
'plan_length': 14,
```

```
'time_in_sec': 3328.9549978948726},
'astar_search with h_unmet_goals': {'actions': 104,
'expansions': 34330,
'goal_tests': 34332,
'new_nodes': 328509,
'plan_length': 14,
'time_in_sec': 8.050648922012897},
'breadth_first_search': {'actions': 104,
'expansions': 99736,
'goal_tests': 114953,
'new_nodes': 944130,
'plan_length': 14,
'time_in_sec': 4.449251639847716},
'depth_first_graph_search': {'actions': 104,
'expansions': 25174,
'goal_tests': 25175,
'new_nodes': 228849,
'plan_length': 24132,
'time_in_sec': 1052.8746864348748},
'greedy_best_first_graph_search with h_pg_levelsum': {'actions': 104,
'expansions': 17,
'goal_tests': 19,
'new_nodes': 165,
'plan_length': 17,
'time_in_sec': 3.4156429592508175},
'greedy_best_first_graph_search with h_pg_maxlevel': {'actions': 104,
'expansions': 56,
'goal_tests': 58,
'new_nodes': 580,
'plan_length': 17,
'time_in_sec': 2.9414996963962494},
'greedy_best_first_graph_search with h_pg_setlevel': {'actions': 104,
'expansions': 107,
'goal_tests': 109,
'new_nodes': 1164,
'plan_length': 23,
'time_in_sec': 29.809604525352142},
'greedy_best_first_graph_search with h_unmet_goals': {'actions': 104,
'expansions': 29,
'goal_tests': 31,
'new_nodes': 280,
'plan_length': 18,
'time_in_sec': 0.033204580924575566},
```

```
'uniform_cost_search': {'actions': 104,  
    'expansions': 113339,  
    'goal_tests': 113341,  
    'new_nodes': 1066413,  
    'plan_length': 14,  
    'time_in_sec': 14.029020421490713}}}
```

```
In [7]: # Ensure the length of list is as expected.  
        # There are 4 problems and 11 search algorithms.  
        # 4 x 11 = 44  
        # There should be 44 elements.  
        len(list_of_dicts_row_oriented)
```

```
Out[7]: 44
```

```
In [8]: list_of_dicts_row_oriented
```

```
Out[8]: [{ 'actions': 20,
           'expansions': 60,
           'goal_tests': 62,
           'new_nodes': 240,
           'plan_length': 6,
           'problem': 'Air Cargo Problem 1',
           'search': 'uniform_cost_search',
           'time_in_sec': 0.01970920338766699},
        { 'actions': 20,
           'expansions': 33,
           'goal_tests': 35,
           'new_nodes': 138,
           'plan_length': 6,
           'problem': 'Air Cargo Problem 1',
           'search': 'astar_search with h_pg_setlevel',
           'time_in_sec': 0.29609892190720943},
        { 'actions': 20,
           'expansions': 6,
           'goal_tests': 8,
           'new_nodes': 28,
           'plan_length': 6,
           'problem': 'Air Cargo Problem 1',
           'search': 'greedy_best_first_graph_search with h_pg_levelsum',
           'time_in_sec': 0.3834521536819968},
        { 'actions': 20,
           'expansions': 6,
           'goal_tests': 8,
           'new_nodes': 28,
           'plan_length': 6,
           'problem': 'Air Cargo Problem 1',
           'search': 'greedy_best_first_graph_search with h_pg_setlevel',
           'time_in_sec': 0.41804252144090803},
        { 'actions': 20,
           'expansions': 21,
           'goal_tests': 22,
           'new_nodes': 84,
           'plan_length': 20,
           'problem': 'Air Cargo Problem 1',
           'search': 'depth_first_graph_search',
           'time_in_sec': 0.006418696626931376},
        { 'actions': 20,
           'expansions': 6,
           'goal_tests': 8,
```

```
'new_nodes': 24,
'plan_length': 6,
'problem': 'Air Cargo Problem 1',
'search': 'greedy_best_first_graph_search with h_pg_maxlevel',
'time_in_sec': 0.08394447825654622},
{'actions': 20,
'expansions': 7,
'goal_tests': 9,
'new_nodes': 29,
'plan_length': 6,
'problem': 'Air Cargo Problem 1',
'search': 'greedy_best_first_graph_search with h_unmet_goals',
'time_in_sec': 0.0026820088196692904},
{'actions': 20,
'expansions': 28,
'goal_tests': 30,
'new_nodes': 122,
'plan_length': 6,
'problem': 'Air Cargo Problem 1',
'search': 'astar_search with h_pg_levelsum',
'time_in_sec': 0.21640066376721034},
{'actions': 20,
'expansions': 43,
'goal_tests': 56,
'new_nodes': 178,
'plan_length': 6,
'problem': 'Air Cargo Problem 1',
'search': 'breadth_first_search',
'time_in_sec': 0.02428653135087065},
{'actions': 20,
'expansions': 43,
'goal_tests': 45,
'new_nodes': 180,
'plan_length': 6,
'problem': 'Air Cargo Problem 1',
'search': 'astar_search with h_pg_maxlevel',
'time_in_sec': 0.12295208562681625},
{'actions': 20,
'expansions': 50,
'goal_tests': 52,
'new_nodes': 206,
'plan_length': 6,
'problem': 'Air Cargo Problem 1',
```

```

    'search': 'astar_search with h_unmet_goals',
    'time_in_sec': 0.015270101490373222},
{'actions': 72,
 'expansions': 5154,
 'goal_tests': 5156,
 'new_nodes': 46618,
 'plan_length': 9,
 'problem': 'Air Cargo Problem 2',
 'search': 'uniform_cost_search',
 'time_in_sec': 0.7676067214540252},
{'actions': 72,
 'expansions': 1037,
 'goal_tests': 1039,
 'new_nodes': 9605,
 'plan_length': 9,
 'problem': 'Air Cargo Problem 2',
 'search': 'astar_search with h_pg_setlevel',
 'time_in_sec': 134.70260450348985},
{'actions': 72,
 'expansions': 9,
 'goal_tests': 11,
 'new_nodes': 86,
 'plan_length': 9,
 'problem': 'Air Cargo Problem 2',
 'search': 'greedy_best_first_graph_search with h_pg_levelsum',
 'time_in_sec': 0.6934766464945943},
{'actions': 72,
 'expansions': 9,
 'goal_tests': 11,
 'new_nodes': 84,
 'plan_length': 9,
 'problem': 'Air Cargo Problem 2',
 'search': 'greedy_best_first_graph_search with h_pg_setlevel',
 'time_in_sec': 1.0814222051712061},
{'actions': 72,
 'expansions': 624,
 'goal_tests': 625,
 'new_nodes': 5602,
 'plan_length': 619,
 'problem': 'Air Cargo Problem 2',
 'search': 'depth_first_graph_search',
 'time_in_sec': 0.5629788986626627},
{'actions': 72,

```



```

    'expansions': 27,
    'goal_tests': 29,
    'new_nodes': 249,
    'plan_length': 9,
    'problem': 'Air Cargo Problem 2',
    'search': 'greedy_best_first_graph_search with h_pg_maxlevel',
    'time_in_sec': 0.6004847080362232},
    {'actions': 72,
     'expansions': 17,
     'goal_tests': 19,
     'new_nodes': 170,
     'plan_length': 9,
     'problem': 'Air Cargo Problem 2',
     'search': 'greedy_best_first_graph_search with h_unmet_goals',
     'time_in_sec': 0.05454464031955597},
    {'actions': 72,
     'expansions': 357,
     'goal_tests': 359,
     'new_nodes': 3426,
     'plan_length': 9,
     'problem': 'Air Cargo Problem 2',
     'search': 'astar_search with h_pg_levelsum',
     'time_in_sec': 16.25889883191922},
    {'actions': 72,
     'expansions': 3343,
     'goal_tests': 4609,
     'new_nodes': 30503,
     'plan_length': 9,
     'problem': 'Air Cargo Problem 2',
     'search': 'breadth_first_search',
     'time_in_sec': 0.3474298061943615},
    {'actions': 72,
     'expansions': 2887,
     'goal_tests': 2889,
     'new_nodes': 26594,
     'plan_length': 9,
     'problem': 'Air Cargo Problem 2',
     'search': 'astar_search with h_pg_maxlevel',
     'time_in_sec': 71.23779185270179},
    {'actions': 72,
     'expansions': 2467,
     'goal_tests': 2469,
     'new_nodes': 22522,

```

```
'plan_length': 9,  
'problem': 'Air Cargo Problem 2',  
'search': 'astar_search with h_unmet_goals',  
'time_in_sec': 0.7492702580525501},  
{ 'actions': 104,  
  'expansions': 113339,  
  'goal_tests': 113341,  
  'new_nodes': 1066413,  
  'plan_length': 14,  
  'problem': 'Air Cargo Problem 4',  
  'search': 'uniform_cost_search',  
  'time_in_sec': 14.029020421490713},  
{ 'actions': 104,  
  'expansions': 22606,  
  'goal_tests': 22608,  
  'new_nodes': 224229,  
  'plan_length': 14,  
  'problem': 'Air Cargo Problem 4',  
  'search': 'astar_search with h_pg_setlevel',  
  'time_in_sec': 3328.9549978948726},  
{ 'actions': 104,  
  'expansions': 17,  
  'goal_tests': 19,  
  'new_nodes': 165,  
  'plan_length': 17,  
  'problem': 'Air Cargo Problem 4',  
  'search': 'greedy_best_first_graph_search with h_pg_levelsum',  
  'time_in_sec': 3.4156429592508175},  
{ 'actions': 104,  
  'expansions': 107,  
  'goal_tests': 109,  
  'new_nodes': 1164,  
  'plan_length': 23,  
  'problem': 'Air Cargo Problem 4',  
  'search': 'greedy_best_first_graph_search with h_pg_setlevel',  
  'time_in_sec': 29.809604525352142},  
{ 'actions': 104,  
  'expansions': 25174,  
  'goal_tests': 25175,  
  'new_nodes': 228849,  
  'plan_length': 24132,  
  'problem': 'Air Cargo Problem 4',  
  'search': 'depth_first_graph_search',
```

```

    'time_in_sec': 1052.8746864348748},
{'actions': 104,
 'expansions': 56,
 'goal_tests': 58,
 'new_nodes': 580,
 'plan_length': 17,
 'problem': 'Air Cargo Problem 4',
 'search': 'greedy_best_first_graph_search with h_pg_maxlevel',
 'time_in_sec': 2.9414996963962494},
{'actions': 104,
 'expansions': 29,
 'goal_tests': 31,
 'new_nodes': 280,
 'plan_length': 18,
 'problem': 'Air Cargo Problem 4',
 'search': 'greedy_best_first_graph_search with h_unmet_goals',
 'time_in_sec': 0.033204580924575566},
{'actions': 104,
 'expansions': 1208,
 'goal_tests': 1210,
 'new_nodes': 12210,
 'plan_length': 15,
 'problem': 'Air Cargo Problem 4',
 'search': 'astar_search with h_pg_levelsum',
 'time_in_sec': 195.75180205181073},
{'actions': 104,
 'expansions': 99736,
 'goal_tests': 114953,
 'new_nodes': 944130,
 'plan_length': 14,
 'problem': 'Air Cargo Problem 4',
 'search': 'breadth_first_search',
 'time_in_sec': 4.449251639847716},
{'actions': 104,
 'expansions': 62077,
 'goal_tests': 62079,
 'new_nodes': 599376,
 'plan_length': 14,
 'problem': 'Air Cargo Problem 4',
 'search': 'astar_search with h_pg_maxlevel',
 'time_in_sec': 2261.118180112795},
{'actions': 104,
 'expansions': 34330,

```

```

'goal_tests': 34332,
'new_nodes': 328509,
'plan_length': 14,
'problem': 'Air Cargo Problem 4',
'search': 'astar_search with h_unmet_goals',
'time_in_sec': 8.050648922012897},
{'actions': 88,
'expansions': 18510,
'goal_tests': 18512,
'new_nodes': 161936,
'plan_length': 12,
'problem': 'Air Cargo Problem 3',
'search': 'uniform_cost_search',
'time_in_sec': 2.316739035534681},
{'actions': 88,
'expansions': 3423,
'goal_tests': 3425,
'new_nodes': 31596,
'plan_length': 12,
'problem': 'Air Cargo Problem 3',
'search': 'astar_search with h_pg_setlevel',
'time_in_sec': 317.41891877596595},
{'actions': 88,
'expansions': 14,
'goal_tests': 16,
'new_nodes': 126,
'plan_length': 14,
'problem': 'Air Cargo Problem 3',
'search': 'greedy_best_first_graph_search with h_pg_levelsum',
'time_in_sec': 2.6483832134316003},
{'actions': 88,
'expansions': 35,
'goal_tests': 37,
'new_nodes': 345,
'plan_length': 17,
'problem': 'Air Cargo Problem 3',
'search': 'greedy_best_first_graph_search with h_pg_setlevel',
'time_in_sec': 9.081586877550762},
{'actions': 88,
'expansions': 408,
'goal_tests': 409,
'new_nodes': 3364,
'plan_length': 392,

```

```
'problem': 'Air Cargo Problem 3',
'search': 'depth_first_graph_search',
'time_in_sec': 0.36805228894607467},
{'actions': 88,
'expansions': 21,
'goal_tests': 23,
'new_nodes': 195,
'plan_length': 13,
'problem': 'Air Cargo Problem 3',
'search': 'greedy_best_first_graph_search with h_pg_maxlevel',
'time_in_sec': 1.2394026095106483},
{'actions': 88,
'expansions': 25,
'goal_tests': 27,
'new_nodes': 230,
'plan_length': 15,
'problem': 'Air Cargo Problem 3',
'search': 'greedy_best_first_graph_search with h_unmet_goals',
'time_in_sec': 0.019154536036438685},
{'actions': 88,
'expansions': 369,
'goal_tests': 371,
'new_nodes': 3403,
'plan_length': 12,
'problem': 'Air Cargo Problem 3',
'search': 'astar_search with h_pg_levelsum',
'time_in_sec': 43.686991657774826},
{'actions': 88,
'expansions': 14663,
'goal_tests': 18098,
'new_nodes': 129625,
'plan_length': 12,
'problem': 'Air Cargo Problem 3',
'search': 'breadth_first_search',
'time_in_sec': 1.3805018029333667},
{'actions': 88,
'expansions': 9580,
'goal_tests': 9582,
'new_nodes': 86312,
'plan_length': 12,
'problem': 'Air Cargo Problem 3',
'search': 'astar_search with h_pg_maxlevel',
'time_in_sec': 240.99240215040265},
```

```
{'actions': 88,  
  'expansions': 7388,  
  'goal_tests': 7390,  
  'new_nodes': 65711,  
  'plan_length': 12,  
  'problem': 'Air Cargo Problem 3',  
  'search': 'astar_search with h_unmet_goals',  
  'time_in_sec': 1.6978325306583315}]
```

## Import Pandas

```
In [9]: import pandas as pd
```

## Dataframe Population Examples.

```
In [10]: # Dataframe test - Row Oriented  
# Example from http://pbpython.com/pandas-list-dict.html  
sales = [{'account': 'Jones LLC', 'Jan': 150, 'Feb': 200, 'Mar': 140},  
          {'account': 'Alpha Co', 'Jan': 200, 'Feb': 210, 'Mar': 215},  
          {'account': 'Blue Inc', 'Jan': 50, 'Feb': 90, 'Mar': 95 }]  
df_row_oriented = pd.DataFrame(sales)
```

```
In [11]: df_row_oriented
```

```
Out[11]:
```

	Feb	Jan	Mar	account
0	200	150	140	Jones LLC
1	210	200	215	Alpha Co
2	90	50	95	Blue Inc

```
In [12]: # Dataframe test - Column Oriented
# Example from http://pbpython.com/pandas-list-dict.html
sales = {'account': ['Jones LLC', 'Alpha Co', 'Blue Inc'],
         'Jan': [150, 200, 50],
         'Feb': [200, 210, 90],
         'Mar': [140, 215, 95]}
df_column_oriented = pd.DataFrame.from_dict(sales)
```

```
In [13]: df_column_oriented
```

```
Out[13]:
```

	Feb	Jan	Mar	account
0	200	150	140	Jones LLC
1	210	200	215	Alpha Co
2	90	50	95	Blue Inc

## Create Pandas Dataframe from analysis data.

```
In [14]: df_a = pd.DataFrame(list_of_dicts_row_oriented)
```

## Export cleaned data as .csv. Load clean data in the future as starting point.

```
In [15]: file_path_and_name = 'data/log_output_all_a.csv'
df_a.to_csv( file_path_and_name,
             index = False )
```

Import cleaned data from file. Can use this as a starting point and skip the cleaning of the data from the raw data collected.

```
In [16]: file_path_and_name = 'data/log_output_all_a.csv'
df_a = pd.read_csv(file_path_and_name,
                    #index_col = 'time_in_sec'
                    index_col = None)
```



```
In [17]: # Check the first 10 entries.
# NOTE: Order is not enforced because it data was populated
#        via an unordered dictionary.
df_a.head(10)
```

Out[17]:

	actions	expansions	goal_tests	new_nodes	plan_length	problem	search	time_in_sec
0	20	60	62	240	6	Air Cargo Problem 1	uniform_cost_search	0.019709
1	20	33	35	138	6	Air Cargo Problem 1	astar_search with h_pg_setlevel	0.296099
2	20	6	8	28	6	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_levelsum	0.383452
3	20	6	8	28	6	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_setlevel	0.418043
4	20	21	22	84	20	Air Cargo Problem 1	depth_first_graph_search	0.006419
5	20	6	8	24	6	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_maxlevel	0.083944
6	20	7	9	29	6	Air Cargo Problem 1	greedy_best_first_graph_search with h_unmet_goals	0.002682
7	20	28	30	122	6	Air Cargo Problem 1	astar_search with h_pg_levelsum	0.216401
8	20	43	56	178	6	Air Cargo Problem 1	breadth_first_search	0.024287
9	20	43	45	180	6	Air Cargo Problem 1	astar_search with h_pg_maxlevel	0.122952

```
In [18]: # Check the last 10 entries.
# NOTE: Order is not enforced because it data was populated
#        via an unordered dictionary.
df_a.tail(10)
```

Out[18]:

	actions	expansions	goal_tests	new_nodes	plan_length	problem	search	time_in_sec
<b>34</b>	88	3423	3425	31596	12	Air Cargo Problem 3	astar_search with h_pg_setlevel	317.418919
<b>35</b>	88	14	16	126	14	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_levelsum	2.648383
<b>36</b>	88	35	37	345	17	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_setlevel	9.081587
<b>37</b>	88	408	409	3364	392	Air Cargo Problem 3	depth_first_graph_search	0.368052
<b>38</b>	88	21	23	195	13	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_maxlevel	1.239403
<b>39</b>	88	25	27	230	15	Air Cargo Problem 3	greedy_best_first_graph_search with h_unmet_goals	0.019155
<b>40</b>	88	369	371	3403	12	Air Cargo Problem 3	astar_search with h_pg_levelsum	43.686992
<b>41</b>	88	14663	18098	129625	12	Air Cargo Problem 3	breadth_first_search	1.380502
<b>42</b>	88	9580	9582	86312	12	Air Cargo Problem 3	astar_search with h_pg_maxlevel	240.992402
<b>43</b>	88	7388	7390	65711	12	Air Cargo Problem 3	astar_search with h_unmet_goals	1.697833

```

In [19]: # Check the random sample of 10 entries.
# NOTE: Using a random_state number (a seed) to generate the
#        same sample each run. To get a different sample each
#        time, use value of None.
# NOTE: Order is not enforced because it data was populated
#        via an unordered dictionary.
df_a.sample( 10,
             random_state = 7 )

```

Out[19]:

	actions	expansions	goal_tests	new_nodes	plan_length	problem	search	time_in_sec
<b>20</b>	72	2887	2889	26594	9	Air Cargo Problem 2	astar_search with h_pg_maxlevel	71.237792
<b>18</b>	72	357	359	3426	9	Air Cargo Problem 2	astar_search with h_pg_levelsum	16.258899
<b>13</b>	72	9	11	86	9	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_levelsum	0.693477
<b>21</b>	72	2467	2469	22522	9	Air Cargo Problem 2	astar_search with h_unmet_goals	0.749270
<b>1</b>	20	33	35	138	6	Air Cargo Problem 1	astar_search with h_pg_setlevel	0.296099
<b>10</b>	20	50	52	206	6	Air Cargo Problem 1	astar_search with h_unmet_goals	0.015270
<b>15</b>	72	624	625	5602	619	Air Cargo Problem 2	depth_first_graph_search	0.562979
<b>17</b>	72	17	19	170	9	Air Cargo Problem 2	greedy_best_first_graph_search with h_unmet_goals	0.054545
<b>24</b>	104	17	19	165	17	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_levelsum	3.415643
<b>41</b>	88	14663	18098	129625	12	Air Cargo Problem 3	breadth_first_search	1.380502

```
In [20]: # Show the entire DataFrame
#
df_a
```

Out[20]:

	actions	expansions	goal_tests	new_nodes	plan_length	problem	search	time_in_sec
<b>0</b>	20	60	62	240	6	Air Cargo Problem 1	uniform_cost_search	0.019709
<b>1</b>	20	33	35	138	6	Air Cargo Problem 1	astar_search with h_pg_setlevel	0.296099
<b>2</b>	20	6	8	28	6	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_levelsum	0.383452
<b>3</b>	20	6	8	28	6	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_setlevel	0.418043
<b>4</b>	20	21	22	84	20	Air Cargo Problem 1	depth_first_graph_search	0.006419
<b>5</b>	20	6	8	24	6	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_maxlevel	0.083944
<b>6</b>	20	7	9	29	6	Air Cargo Problem 1	greedy_best_first_graph_search with h_unmet_goals	0.002682
<b>7</b>	20	28	30	122	6	Air Cargo Problem 1	astar_search with h_pg_levelsum	0.216401
<b>8</b>	20	43	56	178	6	Air Cargo Problem 1	breadth_first_search	0.024287
<b>9</b>	20	43	45	180	6	Air Cargo Problem 1	astar_search with h_pg_maxlevel	0.122952
<b>10</b>	20	50	52	206	6	Air Cargo Problem 1	astar_search with h_unmet_goals	0.015270
<b>11</b>	72	5154	5156	46618	9	Air Cargo Problem 2	uniform_cost_search	0.767607
<b>12</b>	72	1037	1039	9605	9	Air Cargo Problem 2	astar_search with h_pg_setlevel	134.702605
<b>13</b>	72	9	11	86	9	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_levelsum	0.693477

	actions	expansions	goal_tests	new_nodes	plan_length	problem	search	time_in_sec
14	72	9	11	84	9	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_setlevel	1.081422
15	72	624	625	5602	619	Air Cargo Problem 2	depth_first_graph_search	0.562979
16	72	27	29	249	9	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_maxlevel	0.600485
17	72	17	19	170	9	Air Cargo Problem 2	greedy_best_first_graph_search with h_unmet_goals	0.054545
18	72	357	359	3426	9	Air Cargo Problem 2	astar_search with h_pg_levelsum	16.258899
19	72	3343	4609	30503	9	Air Cargo Problem 2	breadth_first_search	0.347430
20	72	2887	2889	26594	9	Air Cargo Problem 2	astar_search with h_pg_maxlevel	71.237792
21	72	2467	2469	22522	9	Air Cargo Problem 2	astar_search with h_unmet_goals	0.749270
22	104	113339	113341	1066413	14	Air Cargo Problem 4	uniform_cost_search	14.029020
23	104	22606	22608	224229	14	Air Cargo Problem 4	astar_search with h_pg_setlevel	3328.954998
24	104	17	19	165	17	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_levelsum	3.415643
25	104	107	109	1164	23	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_setlevel	29.809605
26	104	25174	25175	228849	24132	Air Cargo Problem 4	depth_first_graph_search	1052.874686
27	104	56	58	580	17	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_maxlevel	2.941500

	actions	expansions	goal_tests	new_nodes	plan_length	problem	search	time_in_sec
<b>28</b>	104	29	31	280	18	Air Cargo Problem 4	greedy_best_first_graph_search with h_unmet_goals	0.033205
<b>29</b>	104	1208	1210	12210	15	Air Cargo Problem 4	astar_search with h_pg_levelsum	195.751802
<b>30</b>	104	99736	114953	944130	14	Air Cargo Problem 4	breadth_first_search	4.449252
<b>31</b>	104	62077	62079	599376	14	Air Cargo Problem 4	astar_search with h_pg_maxlevel	2261.118180
<b>32</b>	104	34330	34332	328509	14	Air Cargo Problem 4	astar_search with h_unmet_goals	8.050649
<b>33</b>	88	18510	18512	161936	12	Air Cargo Problem 3	uniform_cost_search	2.316739
<b>34</b>	88	3423	3425	31596	12	Air Cargo Problem 3	astar_search with h_pg_setlevel	317.418919
<b>35</b>	88	14	16	126	14	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_levelsum	2.648383
<b>36</b>	88	35	37	345	17	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_setlevel	9.081587
<b>37</b>	88	408	409	3364	392	Air Cargo Problem 3	depth_first_graph_search	0.368052
<b>38</b>	88	21	23	195	13	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_maxlevel	1.239403
<b>39</b>	88	25	27	230	15	Air Cargo Problem 3	greedy_best_first_graph_search with h_unmet_goals	0.019155
<b>40</b>	88	369	371	3403	12	Air Cargo Problem 3	astar_search with h_pg_levelsum	43.686992
<b>41</b>	88	14663	18098	129625	12	Air Cargo Problem 3	breadth_first_search	1.380502

	actions	expansions	goal_tests	new_nodes	plan_length	problem	search	time_in_sec
<b>42</b>	88	9580	9582	86312	12	Air Cargo Problem 3	astar_search with h_pg_maxlevel	240.992402
<b>43</b>	88	7388	7390	65711	12	Air Cargo Problem 3	astar_search with h_unmet_goals	1.697833

```
In [21]: # Reorder the columns
df_b = df_a[['problem', 'search', 'actions', 'expansions', 'goal_tests', 'new_nodes', 'plan_length',
'time_in_sec']]
```



```
In [22]: # Check the new column order  
df_b
```

Out[22]:

	problem	search	actions	expansions	goal_tests	new_nodes	plan_length	time_in_sec
0	Air Cargo Problem 1	uniform_cost_search	20	60	62	240	6	0.019709
1	Air Cargo Problem 1	astar_search with h_pg_setlevel	20	33	35	138	6	0.296099
2	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_levelsum	20	6	8	28	6	0.383452
3	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_setlevel	20	6	8	28	6	0.418043
4	Air Cargo Problem 1	depth_first_graph_search	20	21	22	84	20	0.006419
5	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_maxlevel	20	6	8	24	6	0.083944
6	Air Cargo Problem 1	greedy_best_first_graph_search with h_unmet_goals	20	7	9	29	6	0.002682
7	Air Cargo Problem 1	astar_search with h_pg_levelsum	20	28	30	122	6	0.216401
8	Air Cargo Problem 1	breadth_first_search	20	43	56	178	6	0.024287
9	Air Cargo Problem 1	astar_search with h_pg_maxlevel	20	43	45	180	6	0.122952
10	Air Cargo Problem 1	astar_search with h_unmet_goals	20	50	52	206	6	0.015270
11	Air Cargo Problem 2	uniform_cost_search	72	5154	5156	46618	9	0.767607
12	Air Cargo Problem 2	astar_search with h_pg_setlevel	72	1037	1039	9605	9	134.702605
13	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_levelsum	72	9	11	86	9	0.693477

	problem	search	actions	expansions	goal_tests	new_nodes	plan_length	time_in_sec
14	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_setlevel	72	9	11	84	9	1.081422
15	Air Cargo Problem 2	depth_first_graph_search	72	624	625	5602	619	0.562979
16	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_maxlevel	72	27	29	249	9	0.600485
17	Air Cargo Problem 2	greedy_best_first_graph_search with h_unmet_goals	72	17	19	170	9	0.054545
18	Air Cargo Problem 2	astar_search with h_pg_levelsum	72	357	359	3426	9	16.258899
19	Air Cargo Problem 2	breadth_first_search	72	3343	4609	30503	9	0.347430
20	Air Cargo Problem 2	astar_search with h_pg_maxlevel	72	2887	2889	26594	9	71.237792
21	Air Cargo Problem 2	astar_search with h_unmet_goals	72	2467	2469	22522	9	0.749270
22	Air Cargo Problem 4	uniform_cost_search	104	113339	113341	1066413	14	14.029020
23	Air Cargo Problem 4	astar_search with h_pg_setlevel	104	22606	22608	224229	14	3328.954998
24	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_levelsum	104	17	19	165	17	3.415643
25	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_setlevel	104	107	109	1164	23	29.809605
26	Air Cargo Problem 4	depth_first_graph_search	104	25174	25175	228849	24132	1052.874686
27	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_maxlevel	104	56	58	580	17	2.941500

	<b>problem</b>	<b>search</b>	<b>actions</b>	<b>expansions</b>	<b>goal_tests</b>	<b>new_nodes</b>	<b>plan_length</b>	<b>time_in_sec</b>
<b>28</b>	Air Cargo Problem 4	greedy_best_first_graph_search with h_unmet_goals	104	29	31	280	18	0.033205
<b>29</b>	Air Cargo Problem 4	astar_search with h_pg_levelsum	104	1208	1210	12210	15	195.751802
<b>30</b>	Air Cargo Problem 4	breadth_first_search	104	99736	114953	944130	14	4.449252
<b>31</b>	Air Cargo Problem 4	astar_search with h_pg_maxlevel	104	62077	62079	599376	14	2261.118180
<b>32</b>	Air Cargo Problem 4	astar_search with h_unmet_goals	104	34330	34332	328509	14	8.050649
<b>33</b>	Air Cargo Problem 3	uniform_cost_search	88	18510	18512	161936	12	2.316739
<b>34</b>	Air Cargo Problem 3	astar_search with h_pg_setlevel	88	3423	3425	31596	12	317.418919
<b>35</b>	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_levelsum	88	14	16	126	14	2.648383
<b>36</b>	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_setlevel	88	35	37	345	17	9.081587
<b>37</b>	Air Cargo Problem 3	depth_first_graph_search	88	408	409	3364	392	0.368052
<b>38</b>	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_maxlevel	88	21	23	195	13	1.239403
<b>39</b>	Air Cargo Problem 3	greedy_best_first_graph_search with h_unmet_goals	88	25	27	230	15	0.019155
<b>40</b>	Air Cargo Problem 3	astar_search with h_pg_levelsum	88	369	371	3403	12	43.686992
<b>41</b>	Air Cargo Problem 3	breadth_first_search	88	14663	18098	129625	12	1.380502

	problem	search	actions	expansions	goal_tests	new_nodes	plan_length	time_in_sec
42	Air Cargo Problem 3	astar_search with h_pg_maxlevel	88	9580	9582	86312	12	240.992402
43	Air Cargo Problem 3	astar_search with h_unmet_goals	88	7388	7390	65711	12	1.697833

## Answering Questions from the rubric

### Experimental Results & Report

Analyze the search complexity as a function of domain size, search algorithm, and heuristic.

Report includes a table or chart to analyze the number of nodes expanded against number of actions in the domain.

The chart or table includes data for all search & heuristic combinations for air cargo problems 1 and 2

The chart or table includes data at least one uninformed search, two heuristics with greedy best first search, and two heuristics with A\* on air cargo problems 3 and 4.

```
In [23]: # This is the entire DataFrame of the 4 problems and 11 search algorithms.
# NOTE: Using the column 'time_in_sec' as the index. It was sorted in ascending order.
column_as_index = 'time_in_sec'
df_c = df_b.sort_values( column_as_index, ascending = True ).set_index( column_as_index)
```

```
In [25]: # View with ascending sorted time_in_sec as index  
df_c
```

Out[25]:

	problem	search	actions	expansions	goal_tests	new_nodes	plan_length
time_in_sec							
0.002682	Air Cargo Problem 1	greedy_best_first_graph_search with h_unmet_goals	20	7	9	29	6
0.006419	Air Cargo Problem 1	depth_first_graph_search	20	21	22	84	20
0.015270	Air Cargo Problem 1	astar_search with h_unmet_goals	20	50	52	206	6
0.019155	Air Cargo Problem 3	greedy_best_first_graph_search with h_unmet_goals	88	25	27	230	15
0.019709	Air Cargo Problem 1	uniform_cost_search	20	60	62	240	6
0.024287	Air Cargo Problem 1	breadth_first_search	20	43	56	178	6
0.033205	Air Cargo Problem 4	greedy_best_first_graph_search with h_unmet_goals	104	29	31	280	18
0.054545	Air Cargo Problem 2	greedy_best_first_graph_search with h_unmet_goals	72	17	19	170	9
0.083944	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_maxlevel	20	6	8	24	6
0.122952	Air Cargo Problem 1	astar_search with h_pg_maxlevel	20	43	45	180	6
0.216401	Air Cargo Problem 1	astar_search with h_pg_levelsum	20	28	30	122	6
0.296099	Air Cargo Problem 1	astar_search with h_pg_setlevel	20	33	35	138	6
0.347430	Air Cargo Problem 2	breadth_first_search	72	3343	4609	30503	9

	<b>problem</b>	<b>search</b>	<b>actions</b>	<b>expansions</b>	<b>goal_tests</b>	<b>new_nodes</b>	<b>plan_length</b>
<b>time_in_sec</b>							
<b>0.368052</b>	Air Cargo Problem 3	depth_first_graph_search	88	408	409	3364	392
<b>0.383452</b>	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_levelsum	20	6	8	28	6
<b>0.418043</b>	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_setlevel	20	6	8	28	6
<b>0.562979</b>	Air Cargo Problem 2	depth_first_graph_search	72	624	625	5602	619
<b>0.600485</b>	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_maxlevel	72	27	29	249	9
<b>0.693477</b>	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_levelsum	72	9	11	86	9
<b>0.749270</b>	Air Cargo Problem 2	astar_search with h_unmet_goals	72	2467	2469	22522	9
<b>0.767607</b>	Air Cargo Problem 2	uniform_cost_search	72	5154	5156	46618	9
<b>1.081422</b>	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_setlevel	72	9	11	84	9
<b>1.239403</b>	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_maxlevel	88	21	23	195	13
<b>1.380502</b>	Air Cargo Problem 3	breadth_first_search	88	14663	18098	129625	12
<b>1.697833</b>	Air Cargo Problem 3	astar_search with h_unmet_goals	88	7388	7390	65711	12
<b>2.316739</b>	Air Cargo Problem 3	uniform_cost_search	88	18510	18512	161936	12



	<b>problem</b>	<b>search</b>	<b>actions</b>	<b>expansions</b>	<b>goal_tests</b>	<b>new_nodes</b>	<b>plan_length</b>
<b>time_in_sec</b>							
<b>2.648383</b>	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_levelsum	88	14	16	126	14
<b>2.941500</b>	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_maxlevel	104	56	58	580	17
<b>3.415643</b>	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_levelsum	104	17	19	165	17
<b>4.449252</b>	Air Cargo Problem 4	breadth_first_search	104	99736	114953	944130	14
<b>8.050649</b>	Air Cargo Problem 4	astar_search with h_unmet_goals	104	34330	34332	328509	14
<b>9.081587</b>	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_setlevel	88	35	37	345	17
<b>14.029020</b>	Air Cargo Problem 4	uniform_cost_search	104	113339	113341	1066413	14
<b>16.258899</b>	Air Cargo Problem 2	astar_search with h_pg_levelsum	72	357	359	3426	9
<b>29.809605</b>	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_setlevel	104	107	109	1164	23
<b>43.686992</b>	Air Cargo Problem 3	astar_search with h_pg_levelsum	88	369	371	3403	12
<b>71.237792</b>	Air Cargo Problem 2	astar_search with h_pg_maxlevel	72	2887	2889	26594	9
<b>134.702605</b>	Air Cargo Problem 2	astar_search with h_pg_setlevel	72	1037	1039	9605	9
<b>195.751802</b>	Air Cargo Problem 4	astar_search with h_pg_levelsum	104	1208	1210	12210	15

	problem	search	actions	expansions	goal_tests	new_nodes	plan_length
time_in_sec							
<b>240.992402</b>	Air Cargo Problem 3	astar_search with h_pg_maxlevel	88	9580	9582	86312	12
<b>317.418919</b>	Air Cargo Problem 3	astar_search with h_pg_setlevel	88	3423	3425	31596	12
<b>1052.874686</b>	Air Cargo Problem 4	depth_first_graph_search	104	25174	25175	228849	24132
<b>2261.118180</b>	Air Cargo Problem 4	astar_search with h_pg_maxlevel	104	62077	62079	599376	14
<b>3328.954998</b>	Air Cargo Problem 4	astar_search with h_pg_setlevel	104	22606	22608	224229	14

**Checking with a line and bar graph, in general, the number of expansions and the resulting number of actions.**

```
In [26]: # Just using the expansions and actions columns
#df_expansion_actions = df_b[['expansions', 'actions']].sort_values('expansions').groupby('expansions').sum()
df_expansion_actions = df_b[['expansions', 'actions']].sort_values('expansions')
```

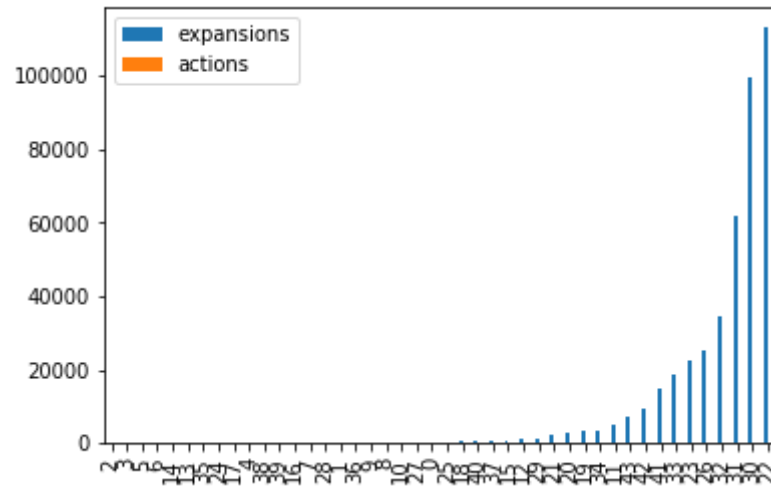
```
In [27]: # Display the DataFrame.  
df_expansion_actions
```

Out[27]:

	expansions	actions
2	6	20
3	6	20
5	6	20
6	7	20
14	9	72
13	9	72
35	14	88
24	17	104
17	17	72
4	21	20
38	21	88
39	25	88
16	27	72
7	28	20
28	29	104
1	33	20
36	35	88
9	43	20
8	43	20
10	50	20
27	56	104
0	60	20
25	107	104

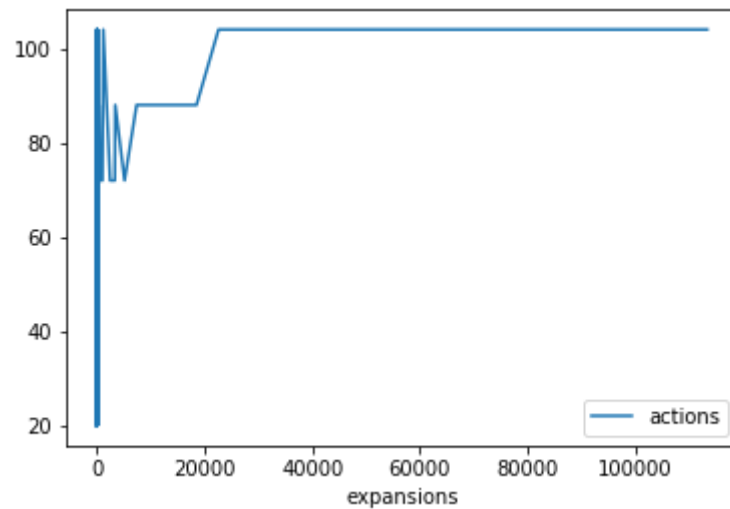
	<b>expansions</b>	<b>actions</b>
<b>18</b>	357	72
<b>40</b>	369	88
<b>37</b>	408	88
<b>15</b>	624	72
<b>12</b>	1037	72
<b>29</b>	1208	104
<b>21</b>	2467	72
<b>20</b>	2887	72
<b>19</b>	3343	72
<b>34</b>	3423	88
<b>11</b>	5154	72
<b>43</b>	7388	88
<b>42</b>	9580	88
<b>41</b>	14663	88
<b>33</b>	18510	88
<b>23</b>	22606	104
<b>26</b>	25174	104
<b>32</b>	34330	104
<b>31</b>	62077	104
<b>30</b>	99736	104
<b>22</b>	113339	104

```
In [29]: # Plot the bar graph
df_expansion_actions.plot(kind='bar');
```



```
In [30]: df_expansion_actions.plot(x='expansions', y='actions', style='-')
```

```
Out[30]: <matplotlib.axes._subplots.AxesSubplot at 0x1128a3400>
```



Checking time in seconds versions expansions

```
In [31]: df_time_actions = df_b[['time_in_sec', 'expansions']]

# Sort the values based on time_in_sec
df_time_actions = df_time_actions.sort_values( 'time_in_sec', ascending = True )
```

```
In [32]: df_time_actions
```

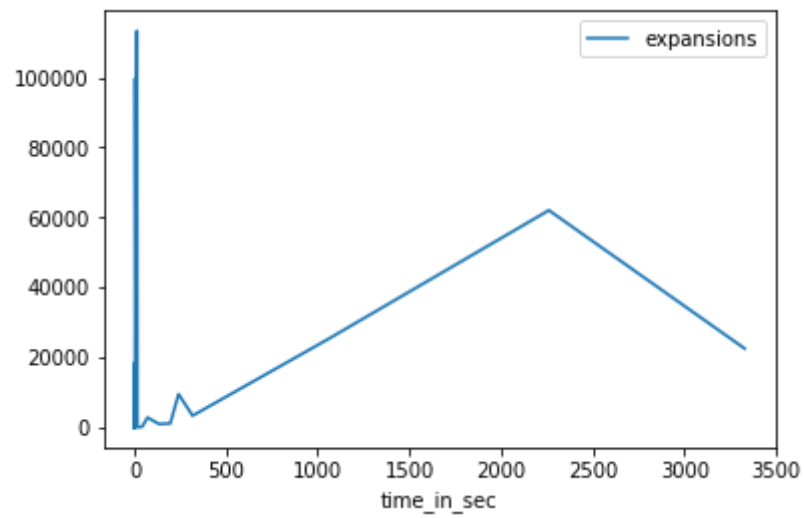


Out [ 32 ] :

	time_in_sec	expansions
6	0.002682	7
4	0.006419	21
10	0.015270	50
39	0.019155	25
0	0.019709	60
8	0.024287	43
28	0.033205	29
17	0.054545	17
5	0.083944	6
9	0.122952	43
7	0.216401	28
1	0.296099	33
19	0.347430	3343
37	0.368052	408
2	0.383452	6
3	0.418043	6
15	0.562979	624
16	0.600485	27
13	0.693477	9
21	0.749270	2467
11	0.767607	5154
14	1.081422	9
38	1.239403	21

	time_in_sec	expansions
41	1.380502	14663
43	1.697833	7388
33	2.316739	18510
35	2.648383	14
27	2.941500	56
24	3.415643	17
30	4.449252	99736
32	8.050649	34330
36	9.081587	35
22	14.029020	113339
18	16.258899	357
25	29.809605	107
40	43.686992	369
20	71.237792	2887
12	134.702605	1037
29	195.751802	1208
42	240.992402	9580
34	317.418919	3423
26	1052.874686	25174
31	2261.118180	62077
23	3328.954998	22606

```
In [33]: df_time_actions.plot(x='time_in_sec', y='expansions', style='-');
```



Report includes at least a one paragraph discussion of these results that analyzes the growth trends as the problem size increases

## My Paragraph:

**Discussion of these results that analyzes the growth trends as the problem size increases in terms of the search time against the number of actions in the domain as well as the number of nodes expanded against number of actions in the domain.**

There are 4 problems. The problem size increases more than the last problem (ascending order 1,2,3,4.) When looking at the DataFrame with all 44 scenarios of 4 problems each solved with 11 search algorithms, we can see that overall the time in seconds to solve the problem increases meaning the problem takes less time to solve when compared against the next problem in ascending order. The relationships is clearly not linear. We have dynamic range of

time in seconds 0.002682 for Air Cargo Problem 1 greedy\_best\_first\_graph\_search with h\_unmet\_goals

to

time in seconds 3328.954998 for astar\_search with h\_pg\_setlevel

The number of actions do not necessarily correspond to the number of expansions and vice-versa. For example

Air Cargo Problem 3 greedy\_best\_first\_graph\_search with h\_pg\_maxlevel had 88 actions but 21 expansions.

where as

Air Cargo Problem 3 uniform\_cost\_search also had 88 actions but 18510 expansion

The increase in time seems more exponential as problem size increase with action domain being larger. What seems to have more of an affect on the time to get a solution, however, is the search and heuristic used.

An example would be:

Air Cargo Problem 4 greedy\_best\_first\_graph\_search with h\_unmet\_goals time in seconds 0.033205

compared against

Air Cargo Problem 4 astar\_search with h\_pg\_setlevel time in seconds 3328.954998

## Short answers

**Q:** Which algorithm or algorithms would be most appropriate for planning in a very restricted domain (i.e., one that has only a few actions) and needs to operate in real time?

**A:** If the measure to determine the action with the least options (restricted domain) and operate solve as quickly as possible, by sorting the DataFrame we see:

greedy\_best\_first\_graph\_search with h\_unmet\_goals

Check index 0 DataFrame that follows for the information.

0 20 Air Cargo Problem 1 greedy best first graph search with h unmet a... 7 9 29 6 0.002682

```
In [34]: # Sort by actions primarily, then by time in seconds
df_b.sort_values( ['actions', 'time_in_sec'] ).set_index( 'actions').reset_index()
```

Out [ 34 ] :

	actions	problem	search	expansions	goal_tests	new_nodes	plan_length	time_in_sec
0	20	Air Cargo Problem 1	greedy_best_first_graph_search with h_unmet_goals	7	9	29	6	0.002682
1	20	Air Cargo Problem 1	depth_first_graph_search	21	22	84	20	0.006419
2	20	Air Cargo Problem 1	astar_search with h_unmet_goals	50	52	206	6	0.015270
3	20	Air Cargo Problem 1	uniform_cost_search	60	62	240	6	0.019709
4	20	Air Cargo Problem 1	breadth_first_search	43	56	178	6	0.024287
5	20	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_maxlevel	6	8	24	6	0.083944
6	20	Air Cargo Problem 1	astar_search with h_pg_maxlevel	43	45	180	6	0.122952
7	20	Air Cargo Problem 1	astar_search with h_pg_levelsum	28	30	122	6	0.216401
8	20	Air Cargo Problem 1	astar_search with h_pg_setlevel	33	35	138	6	0.296099
9	20	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_levelsum	6	8	28	6	0.383452
10	20	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_setlevel	6	8	28	6	0.418043
11	72	Air Cargo Problem 2	greedy_best_first_graph_search with h_unmet_goals	17	19	170	9	0.054545
12	72	Air Cargo Problem 2	breadth_first_search	3343	4609	30503	9	0.347430
13	72	Air Cargo Problem 2	depth_first_graph_search	624	625	5602	619	0.562979

	actions	problem	search	expansions	goal_tests	new_nodes	plan_length	time_in_sec
14	72	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_maxlevel	27	29	249	9	0.600485
15	72	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_levelsum	9	11	86	9	0.693477
16	72	Air Cargo Problem 2	astar_search with h_unmet_goals	2467	2469	22522	9	0.749270
17	72	Air Cargo Problem 2	uniform_cost_search	5154	5156	46618	9	0.767607
18	72	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_setlevel	9	11	84	9	1.081422
19	72	Air Cargo Problem 2	astar_search with h_pg_levelsum	357	359	3426	9	16.258899
20	72	Air Cargo Problem 2	astar_search with h_pg_maxlevel	2887	2889	26594	9	71.237792
21	72	Air Cargo Problem 2	astar_search with h_pg_setlevel	1037	1039	9605	9	134.702605
22	88	Air Cargo Problem 3	greedy_best_first_graph_search with h_unmet_goals	25	27	230	15	0.019155
23	88	Air Cargo Problem 3	depth_first_graph_search	408	409	3364	392	0.368052
24	88	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_maxlevel	21	23	195	13	1.239403
25	88	Air Cargo Problem 3	breadth_first_search	14663	18098	129625	12	1.380502
26	88	Air Cargo Problem 3	astar_search with h_unmet_goals	7388	7390	65711	12	1.697833
27	88	Air Cargo Problem 3	uniform_cost_search	18510	18512	161936	12	2.316739



	actions	problem	search	expansions	goal_tests	new_nodes	plan_length	time_in_sec
28	88	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_levelsum	14	16	126	14	2.648383
29	88	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_setlevel	35	37	345	17	9.081587
30	88	Air Cargo Problem 3	astar_search with h_pg_levelsum	369	371	3403	12	43.686992
31	88	Air Cargo Problem 3	astar_search with h_pg_maxlevel	9580	9582	86312	12	240.992402
32	88	Air Cargo Problem 3	astar_search with h_pg_setlevel	3423	3425	31596	12	317.418919
33	104	Air Cargo Problem 4	greedy_best_first_graph_search with h_unmet_goals	29	31	280	18	0.033205
34	104	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_maxlevel	56	58	580	17	2.941500
35	104	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_levelsum	17	19	165	17	3.415643
36	104	Air Cargo Problem 4	breadth_first_search	99736	114953	944130	14	4.449252
37	104	Air Cargo Problem 4	astar_search with h_unmet_goals	34330	34332	328509	14	8.050649
38	104	Air Cargo Problem 4	uniform_cost_search	113339	113341	1066413	14	14.029020
39	104	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_setlevel	107	109	1164	23	29.809605
40	104	Air Cargo Problem 4	astar_search with h_pg_levelsum	1208	1210	12210	15	195.751802
41	104	Air Cargo Problem 4	depth_first_graph_search	25174	25175	228849	24132	1052.874686

	actions	problem	search	expansions	goal_tests	new_nodes	plan_length	time_in_sec
42	104	Air Cargo Problem 4	astar_search with h_pg_maxlevel	62077	62079	599376	14	2261.118180
43	104	Air Cargo Problem 4	astar_search with h_pg_setlevel	22606	22608	224229	14	3328.954998

**Q: Which algorithm or algorithms would be most appropriate for planning in very large domains (e.g., planning delivery routes for all UPS drivers in the U.S. on a given day)**

**A: By looking at sorted DataFrame in descending order with sorting of descending time, looking for the most actions with the least amount of time.**

**greedy\_best\_first\_graph\_search with h\_unmet\_goals**

**10 104 Air Cargo Problem 4 greedy\_best\_first\_graph\_search with h\_unmet\_g... 29 31 280 18 0.033205**

**Check index 10 DataFrame that follows for the information.**

```
In [35]: # Sort by actions primarily, then by time in seconds in DESCENDING order.  
df_b.sort_values( ['actions', 'time_in_sec'], ascending = [False, False] ).set_index( 'actions').reset_index()
```

Out [ 35 ] :

	actions	problem	search	expansions	goal_tests	new_nodes	plan_length	time_in_sec
0	104	Air Cargo Problem 4	astar_search with h_pg_setlevel	22606	22608	224229	14	3328.954998
1	104	Air Cargo Problem 4	astar_search with h_pg_maxlevel	62077	62079	599376	14	2261.118180
2	104	Air Cargo Problem 4	depth_first_graph_search	25174	25175	228849	24132	1052.874686
3	104	Air Cargo Problem 4	astar_search with h_pg_levelsum	1208	1210	12210	15	195.751802
4	104	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_setlevel	107	109	1164	23	29.809605
5	104	Air Cargo Problem 4	uniform_cost_search	113339	113341	1066413	14	14.029020
6	104	Air Cargo Problem 4	astar_search with h_unmet_goals	34330	34332	328509	14	8.050649
7	104	Air Cargo Problem 4	breadth_first_search	99736	114953	944130	14	4.449252
8	104	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_levelsum	17	19	165	17	3.415643
9	104	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_maxlevel	56	58	580	17	2.941500
10	104	Air Cargo Problem 4	greedy_best_first_graph_search with h_unmet_goals	29	31	280	18	0.033205
11	88	Air Cargo Problem 3	astar_search with h_pg_setlevel	3423	3425	31596	12	317.418919
12	88	Air Cargo Problem 3	astar_search with h_pg_maxlevel	9580	9582	86312	12	240.992402
13	88	Air Cargo Problem 3	astar_search with h_pg_levelsum	369	371	3403	12	43.686992

	actions	problem	search	expansions	goal_tests	new_nodes	plan_length	time_in_sec
14	88	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_setlevel	35	37	345	17	9.081587
15	88	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_levelsum	14	16	126	14	2.648383
16	88	Air Cargo Problem 3	uniform_cost_search	18510	18512	161936	12	2.316739
17	88	Air Cargo Problem 3	astar_search with h_unmet_goals	7388	7390	65711	12	1.697833
18	88	Air Cargo Problem 3	breadth_first_search	14663	18098	129625	12	1.380502
19	88	Air Cargo Problem 3	greedy_best_first_graph_search with h_pg_maxlevel	21	23	195	13	1.239403
20	88	Air Cargo Problem 3	depth_first_graph_search	408	409	3364	392	0.368052
21	88	Air Cargo Problem 3	greedy_best_first_graph_search with h_unmet_goals	25	27	230	15	0.019155
22	72	Air Cargo Problem 2	astar_search with h_pg_setlevel	1037	1039	9605	9	134.702605
23	72	Air Cargo Problem 2	astar_search with h_pg_maxlevel	2887	2889	26594	9	71.237792
24	72	Air Cargo Problem 2	astar_search with h_pg_levelsum	357	359	3426	9	16.258899
25	72	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_setlevel	9	11	84	9	1.081422
26	72	Air Cargo Problem 2	uniform_cost_search	5154	5156	46618	9	0.767607
27	72	Air Cargo Problem 2	astar_search with h_unmet_goals	2467	2469	22522	9	0.749270

	actions	problem	search	expansions	goal_tests	new_nodes	plan_length	time_in_sec
28	72	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_levelsum	9	11	86	9	0.693477
29	72	Air Cargo Problem 2	greedy_best_first_graph_search with h_pg_maxlevel	27	29	249	9	0.600485
30	72	Air Cargo Problem 2	depth_first_graph_search	624	625	5602	619	0.562979
31	72	Air Cargo Problem 2	breadth_first_search	3343	4609	30503	9	0.347430
32	72	Air Cargo Problem 2	greedy_best_first_graph_search with h_unmet_goals	17	19	170	9	0.054545
33	20	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_setlevel	6	8	28	6	0.418043
34	20	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_levelsum	6	8	28	6	0.383452
35	20	Air Cargo Problem 1	astar_search with h_pg_setlevel	33	35	138	6	0.296099
36	20	Air Cargo Problem 1	astar_search with h_pg_levelsum	28	30	122	6	0.216401
37	20	Air Cargo Problem 1	astar_search with h_pg_maxlevel	43	45	180	6	0.122952
38	20	Air Cargo Problem 1	greedy_best_first_graph_search with h_pg_maxlevel	6	8	24	6	0.083944
39	20	Air Cargo Problem 1	breadth_first_search	43	56	178	6	0.024287
40	20	Air Cargo Problem 1	uniform_cost_search	60	62	240	6	0.019709
41	20	Air Cargo Problem 1	astar_search with h_unmet_goals	50	52	206	6	0.015270

	actions	problem	search	expansions	goal_tests	new_nodes	plan_length	time_in_sec
42	20	Air Cargo Problem 1	depth_first_graph_search	21	22	84	20	0.006419
43	20	Air Cargo Problem 1	greedy_best_first_graph_search with h_unmet_goals	7	9	29	6	0.002682

**Q: Which algorithm or algorithms would be most appropriate for planning problems where it is important to find only optimal plans?**

**A: I took optimal plan to mean to have the smallest plan length. Then the next criteria would be to have the less amount of time.**

**breadth\_first\_search**

**0 14 Air Cargo Problem 4 breadth\_first\_search 104 99736 114953 944130 4.449252**

**Check index 0 DataFrame that follows for the information.**

```
In [38]: # Mask, only looking at problem 4 because it has the biggest problem size.
df_b_prob4_only = df_b[df_b['problem'] == 'Air Cargo Problem 4']

# Sort by plan_length primarily, then by time in seconds in ASCENDING order.
df_b_prob4_only.sort_values( ['plan_length', 'time_in_sec'], ascending = [True, True] ).set_index( 'plan_length' ).reset_index()
```

Out[38]:

	plan_length	problem	search	actions	expansions	goal_tests	new_nodes	time_in_sec
0	14	Air Cargo Problem 4	breadth_first_search	104	99736	114953	944130	4.449252
1	14	Air Cargo Problem 4	astar_search with h_unmet_goals	104	34330	34332	328509	8.050649
2	14	Air Cargo Problem 4	uniform_cost_search	104	113339	113341	1066413	14.029020
3	14	Air Cargo Problem 4	astar_search with h_pg_maxlevel	104	62077	62079	599376	2261.118180
4	14	Air Cargo Problem 4	astar_search with h_pg_setlevel	104	22606	22608	224229	3328.954998
5	15	Air Cargo Problem 4	astar_search with h_pg_levelsum	104	1208	1210	12210	195.751802
6	17	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_maxlevel	104	56	58	580	2.941500
7	17	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_levelsum	104	17	19	165	3.415643
8	18	Air Cargo Problem 4	greedy_best_first_graph_search with h_unmet_goals	104	29	31	280	0.033205
9	23	Air Cargo Problem 4	greedy_best_first_graph_search with h_pg_setlevel	104	107	109	1164	29.809605
10	24132	Air Cargo Problem 4	depth_first_graph_search	104	25174	25175	228849	1052.874686