Written Analysis

Optimal Plans
ALL TAKEN FROM BFS
PROBLEM 1:
Load(C2, P2, JFK)
Load(C1, P1, SFO)
Fly(P2, JFK, SFO)
Load(C2, P2, SFO)
Fly(P1, SFO, JFK)
Load(C1, P1, JFK)

PROBLEM 2:
Load(C2, P2, JFK)
Load(C1, P1, SFO)
Load(C3, P3, ATL)
Fly(P2, JFK, SFO)
Load(C2, P2, SFO)
Fly(P1, SFO, JFK)
Load(C1, P1, JFK)
Fly(P3, ATL, SFO)
Load(C3, P3, SFO)

PROBLEM 3:
Load(C2, P2, JFK)
Load(C1, P1, SFO)
Fly(P2, JFK, ORD)
Load(C4, P2, ORD)
Fly(P1, SFO, ATL)
Load(C3, P1, ATL)
Fly(P1, ATL, JFK)
Load(C1, P1, JFK)
Load(C3, P1, JFK)
Fly(P2, ORD, SFO)
Load(C2, P2, SFO)
Load(C4, P2, SFO)

Result for non-heuristic search:

Per [1] section 3.4, search strategies that come under the heading of uninformed search (a.k.a., blind search) have no additional information about states beyond that provided in the problem definition. All they can do is generate successors and distinguish a goal state from a non-goal state.

PROBLEM 1

	breadth_first_search	breadth_first_tree_search	depth_first_graph_search	depth_limited_search	uniform_cost_search
Expansions	43	1458	12	101	55
Goal Tests	56	1459	13	271	57
Time elapsed	0.033198938 99606541	1.049229473 1084257	0.009058805 997483432	0.116782741 85396731	0.083431082 07918704
Plan length	6	6	12	50	6

PROBLEM 2

	breadth_first_search	breadth_first_tree_search	depth_first_graph_search	depth_limited_search	uniform_cost_search
Expansions	3343	---	582	222719	4853
Goal Tests	4609	---	583	2053741	4855
Time elapsed	14.89696321 5986034	---	3.392203800 00188	1014.067314 1360749	24.15560687 4031946
Plan length	9	---	575	50	9

PROBLEM 3

	breadth_first_search	breadth_first_tree_search	depth_first_graph_search	depth_limited_search	uniform_cost_search
Expansions	14663	---	627	---	18151
Goal Tests	18098	---	628	---	18153
Time elapsed	114.4821847 6600596	---	3.722325151 0004047	---	57.45030542 6027626
Plan length	12	---	596	---	12

In PROBLEM 1 ,there three ones has the same shortest Plan length:BFS,BFTS,UCS.BFS has the least Expansions Goal Tests and Time elapsed,so the optimal plan is BFS.

In PROBLEM 2, here two ones has the same shortest Plan length:BFS,UCS.BFS has the least Expansions Goal Tests and Time elapsed,so the optimal plan is BFS.

In PROBLEM 3, here two ones has the shortest same Plan length:BFS,UCS.BFS has the least Expansions Goal Tests and Time elapsed,so the optimal plan is BFS.

If finding the optimal path length is critical, what strategy should we use? Because it performs faster than Uniform Cost Search, Breadth First Search is the recommended search strategy. This isn't much of a surprise, as BFS is complete and optimal.As shown in [1] section3.4.7:

Criterion	Breadth-First	Uniform-Cost	Depth-First	Depth-Limited	Iterative Deepening	Bidirectional (if applicable)
Complete?	Yes[a]	Yes[a,b]	No	No	Yes[a]	Yes[a,d]
Time	$O(b^d)$	$O(b^{1+\lfloor C^*/\epsilon \rfloor})$	$O(b^m)$	$O(b^\ell)$	$O(b^d)$	$O(b^{d/2})$
Space	$O(b^d)$	$O(b^{1+\lfloor C^*/\epsilon \rfloor})$	$O(bm)$	$O(b\ell)$	$O(bd)$	$O(b^{d/2})$
Optimal?	Yes[c]	Yes	No	No	Yes[c]	Yes[c,d]

Figure 3.21 Evaluation of tree-search strategies. b is the branching factor; d is the depth of the shallowest solution; m is the maximum depth of the search tree; l is the depth limit. Superscript caveats are as follows: [a] complete if b is finite; [b] complete if step costs $\geq \epsilon$ for positive ϵ; [c] optimal if step costs are all identical; [d] if both directions use breadth-first search.

Which search strategy should we use, if having an optimal path length is not the primary criteria? For problems 2 and 3, DFS has the least Time elapsed,however has more longer Plan length.If you wanna get a plan quickly,you can choose this algorithm.

Result for heuristic search:

PROBELM 1

	recursive_best_first_search	greedy_best_first_graph_search	astar_search	astar_search h_ignore_preconditions	astar_search h_pg_levelsum
Expansions	4229	7	55	41	60
Goal Tests	4230	9	57	43	62
Time elapsed	3.0680696859490126	0.0059218211099505424	0.04932075506076217	0.043797663995064795	0.6410302770091221
Plan length	6	6	6	6	6

PROBLEM2

	recursive_best_first_search	greedy_best_first_graph_search	astar_search	astar_search h_ignore_preconditions	astar_search h_pg_levelsum
Expansions	---	998	4853	1450	5446
Goal Tests	---	1000	4855	1452	5448
Time elapsed	---	2.6729854368604472	13.44924579304643	4.634387897909619	305.66087429004256
Plan length	---	17	9	9	9

PROBLEM3

	recursive_best_first_search	greedy_best_first_graph_search	astar_search	astar_search h_ignore_preconditions	astar_search h_pg_levelsum

Expansions	---	5398	18151	5038	20457
Goal Tests	---	5400	18153	5040	20459
Time elapsed	---	17.157925026956946	58.06719737092499	18.117981372983195	1587.4482484669425
Plan length	---	26	12	12	12

In PROBLEM 1 ,there five ones has the same shortest Plan length:

RBFS,GBFGS,A*(ASH1),A*(ASHIP),A*(ASHPGL).GBFGS has the least Expansions Goal Tests and Time elapsed.In A* algorithm,ASHIP has has the least Expansions Goal Tests and Time elapsed.

In PROBLEM 2 ,there three ones has the same shortest Plan length:

A*(ASH1),A*(ASHIP),A*(ASHPGL).A*(ASHIP) has the least Expansions Goal Tests and Time elapsed.

In PROBLEM 2 ,there three ones has the same shortest Plan length:

A*(ASH1),A*(ASHIP),A*(ASHPGL).A*(ASHIP) has the least Expansions Goal Tests and Time elapsed.

What was the best heuristic used in these problems? Was it better than non-heuristic search planning methods for all problems? Why or why not?

A*(ASHIP) is the best heuristic in these problems.Yes,It is better than all non-heuristic search planning methods.It drop some pre conditions,so i will got a better time elapsed.