Problem 1 **20 Actions**

Search Strategy		Optimal	Path Length	Execution Time(s)	Node Expansions
breadth_first_search		Yes	6	0.006231315	43
depth_first_graph_search		No	20	0.004546048	21
uniform_cost_search		Yes	6	0.010774834	60
Greedy Best First Graph Search	h_unmet_goals	Yes	6	0.001669836	7
	h_pg_levelsum	Yes	6	0.1937945	6
	h_pg_maxlevel	Yes	6	0.135950023	6
	h_pg_setlevel	Yes	6	0.579976024	6
A Star Search	h_unmet_goals	Yes	6	0.009482626	50
	h_pg_levelsum	Yes	6	0.464069416	28
	h_pg_maxlevel	Yes	6	0.484867244	43
	h_pg_setlevel	Yes	6	1.34838309	33

Problem 2 **72 Actions**

Search Strategy		Optimal	Path Length	Execution Time(s)	Node Expansions
breadth_first_search		Yes	9	2.027296406	3343
depth_first_graph_search		No	619	3.103898274	624
uniform_cost_search		Yes	9	3.374556757	5154
Greedy Best First Graph Search	h_unmet_goals	Yes	9	0.019245303	17
	h_pg_levelsum	Yes	9	4.01426328	9
	h_pg_maxlevel	Yes	9	6.185265508	27
	h_pg_setlevel	Yes	9	14.29760745	9
A Star Search	h_unmet_goals	Yes	9	2.286448086	2467
	h_pg_levelsum	Yes	9	108.0970256	357
	h_pg_maxlevel	Yes	9	1815.446513	2887
	h_pg_setlevel	Yes	9	3017.337984	1037

Problem 3 **88 Actions**

Search Strategy		Optimal	Path Length	Execution Time(s)	Node Expansions
breadth_first_search		Yes	12	22.30956693	14663
depth_first_graph_search		No	392	2.434812857	408
uniform_cost_search		Yes	12	30.9665716	18510
Greedy Best First Graph Search	h_unmet_goals	No	15	0.118760799	25
	h_pg_levelsum	No	14	29.87744058	14
A Star Search	h_unmet_goals	Yes	12	8.656603537	7388
	h_pg_levelsum	Yes	12	311.3540165	369

Problem 4 **104 Actions**

Search Strategy		Optimal	Path Length	Execution Time(s)	Node Expansions
breadth_first_search		Yes	14	194.109759	99736
uniform_cost_search		Yes	14	179.3828497	113339
Greedy Best First Graph Search	h_unmet_goals	No	18	0.2687997	29
	h_pg_levelsum	No	17	56.70644695	17
A Star Search	h_unmet_goals	Yes	14	148.8859127	34330
	h_pg_levelsum	No	15	2131.323861	1208

Discussion of results:

For problems 1 and 2, uninformed search heuristics have less execution time but more node expansions than Greedy Best First Graph Search.

Across the problems, Greedy Best First Graph Search has much lesser node expansions and significant less execution time. A Star Search h_pg_levelsum has the highest execution time across the problems.

For problems 3 and 4, A Star Search h_unmet_goals is able to provide optimal path and it's execution is slightly longer than it's Greedy Best First Graph Search counterpart. Breadth First Search works in all problems, but it's execution time increases with complexity.

Questions:

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?

Looking at the results, Breadth First Search would be most appropriate where we have only a few actions. This algorithm has less execution time and lesser node expansions than uniform cost search. This would help in real-time operations.

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 star algorithm with the right heuristic having optimum node expansions and execution time can be considered for planning in very large domains.

In problems 3 and 4, we get to optimal plan with A Star Search h_unmet_goals with lesser execution time and optimum node expansions.

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

If execution time is not a constraint, Breadth First Search would be able provide optimal plans even though it has significant number of node expansions. Uniform Cost Search would be the second option as it has even larger number of node expansions and more execution time. Depth first graph search can be avoided in such scenarios as we didn't get optimal paths across the four problems with it's usage.