Problem 1 result:

	Action	Expansions	Goal Tests	New Nodes	Planning Length	Time
breadth_first_search	20	43	56	178	6	0.00554826
depth_first_graph_search	20	21	22	84	20	0.003148959
uniform_cost_search	20	60	62	240	6	0.009351949
greedy_best_first_graph_search h_unmet_goals	20	7	9	29	6	0.00144932
greedy_best_first_graph_search h_pg_levelsum	20	6	8	28	6	0.21759787
greedy_best_first_graph_search h_pg_maxlevel	20	6	8	24	6	0.165331754
greedy_best_first_graph_search h_pg_setlevel	20	6	8	28	6	0.32477297
astar_search h_unmet_goals	20	50	52	206	6	0.009373139
astar_search h_pg_levelsum	20	28	30	122	6	0.552860418
astar_search h_pg_maxlevel	20	43	45	180	6	0.548409189
astar_search h_pg_setlevel	20	33	35	138	6	0.744218051

Problem 2 result:

	Action	Expansions	Goal Tests	New Nodes	Planning Length	Time
breadth_first_search	72	3343	4609	30503	9	1.822035811
depth_first_graph_search	72	624	625	5602	619	2.754512565
uniform_cost_search	72	5154	5156	46618	9	3.155247217
greedy_best_first_graph_search h_unmet_goals	72	17	19	170	9	0.018100668
greedy_best_first_graph_search h_pg_levelsum	72	9	11	86	9	4.635620461
greedy_best_first_graph_search h_pg_maxlevel	72	27	29	249	9	9.240895273
greedy_best_first_graph_search h_pg_setlevel	72	9	11	84	9	7.372021537
astar_search h_unmet_goals	72	2467	2469	22522	9	2.144628052
astar_search h_pg_levelsum	72	357	359	3426	9	118.946675

astar_search h_pg_maxlevel	72	2887	2889	26594	9	668.4443971
astar_search h_pg_setlevel	72	1037	1039	9605	9	703.2097493

Conclusion about problem 1 and problem 2

- Depth_first_graph_search algorithm is longer plan length than other search algorithm, so it is not optimal.
- Uniform_cost_search have more node expansions, it use more memory so it is not optimal too.
- Greedy_best_first_graph_search, greedy_best_first_graph_search with h_pg_maxlevel have lmore node expansions and longer search time.
- Astar_search h_pg_maxlevel and astar_search h_pg_setlevel is longer search time than other astar_search algorithms => the depth_first_graph_search, uniform_cost_search, greedy_best_first_graph_search with h_pg_maxlevel, astar_search_unmet goald and astar_search_h_pg_levelsum for solving problem 3 and 4

Problem 3 result:

	Action	Expansions	Goal Tests	New Nodes	Planning Length	Time
breadth_first_search	88	14663	18098	129625	12	9.833991568
greedy_best_first_graph_search h_unmet_goals	88	25	27	230	15	0.034121127
greedy_best_first_graph_search h_pg_levelsum	88	14	16	126	14	10.14058212
greedy_best_first_graph_search h_pg_maxlevel	88	21	23	195	13	13.10256988
astar_search h_unmet_goals	88	7388	7390	65711	12	7.944454483
astar_search h_pg_levelsum	88	369	371	3403	12	189.355106

Problem 4 result:

	Action	Expansions	Goal Tests	New Nodes	Planning Length	Time
breadth_first_search	104	99736	114953	944130	14	88.65570505
greedy_best_first_graph_search h_unmet_goals	104	29	31	280	18	0.053890134
greedy_best_first_graph_search h_pg_levelsum	104	17	19	165	17	18.40303106

greedy_best_first_graph_search h_pg_maxlevel	104	56	58	580	17	45.73567961
astar_search h_unmet_goals	104	34330	34332	328509	14	51.65323365
astar_search h_pg_levelsum	104	1208	1210	12210	15	1019.274712

Conclusion about problem 3 and problem 4:

- breadth_first_search, astar_search with h_unmet_goals and astar_search with h_pg_levelsum have short planning but the number new node is too high
- astar_search with h_pg_levelsum look too good but the time is very high
- greedy_best_first_graph_search h_unmet_goals and greedy_best_first_graph_search h_pg_levelsum have short time but planning is high
- greedy_best_first_graph_search h_pg_levelsum have most little node expansion but planning is too high
- greedy_best_first_graph_search alwayls take most little node expansion

Answear 3 question

Q1: 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?

breadth_first_search and greedy_best_first_graph_search with h_unmet_goals seem is good algorithms for planning in a very restricted domain

Q2: 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* search h_pg_levelsum seems to be a good algorithm because its have atleast node expansions than other algorithm.

Q3: Which algorithm or algorithms would be most appropriate for planning problems where it is important to find only optimal plans?

breadth_first_search can be a good algorithm for planning problem optimal