William Burkes
CSC412
Assignment 2
3rd Edition

3.23

	Action	Open Nodes	f(n)	g(n)	h(n)
1	Initial State	(L,244)	244	0	244
2	Expand Lugoj				
	Add Timisoara	(T,440)	440	111	329
	Add Mehadia	(T,440),(M,301)	310	70	241
3	Expand Mehadia				
	Add Drobeta	(T,440),(D,387)	387	145	242
4	Expand Drobeta				
	Add Craiova	(T,440),(C,425)	425	265	160
5	Expand Craiova				
	Add Rimmicu Vilcea	(T,440),(R,604)	604	411	193
	Add Pitesti	(T,440),(R,604),(P,503)	503	403	100
6	Expand Timisoara				
	Add Arad	(R,604),(P,503),(A,595)	595	229	366
7	Expand Pitesti				
	Add Bucharest	(R,604),(A,595),(B,504)	504	504	0
8	Expand Bucharest				
	At Goal State	(R,604),(A,595)	-	-	-

3.25

Complete whenever $0 \leq w \leq 2$.
w = 0 gives f(n) = 2g(n). This is the same behavior as uniform-cost search
w = 1 gives the A* search
w = 2 gives f(n) = 2h(n). This is the same as greedy best-first search

$$f(n) = (2 - w)[g(n) + \frac{w}{2-w} h(n)]$$

behaves as A* with a heuristic $\frac{w}{2-w} h(n)$

3.21

a) When all steps cost equal, g(n) ∞ depth(n) so uniform-cost search reproduces breadth-first search

b) Breadth first search is best first search with f(n) = depth(n)
Depth first is best first with f(n) = -depth(n)
Uniform cost is best first with f(n) = g(n)

c) Uniform cost search is A* with h(n) = 0

3.30

a) The TSP problem is to find the shortest path through the cities that form a closed loop. MST finds the shortest path but doesn't need to be a closed loop. MST is admissible because it is always shorter than or equal to a closed loop.

b) Straight line distance back to the start city isn't a good heuristic. It doesn't do well when there are a lot of cities. In the late stages of a search when there aren't many cities left it is more viable. MST dominates straight line distance because it includes the goal node and the current node must either be the straight line between them or include two or more lines that add to more