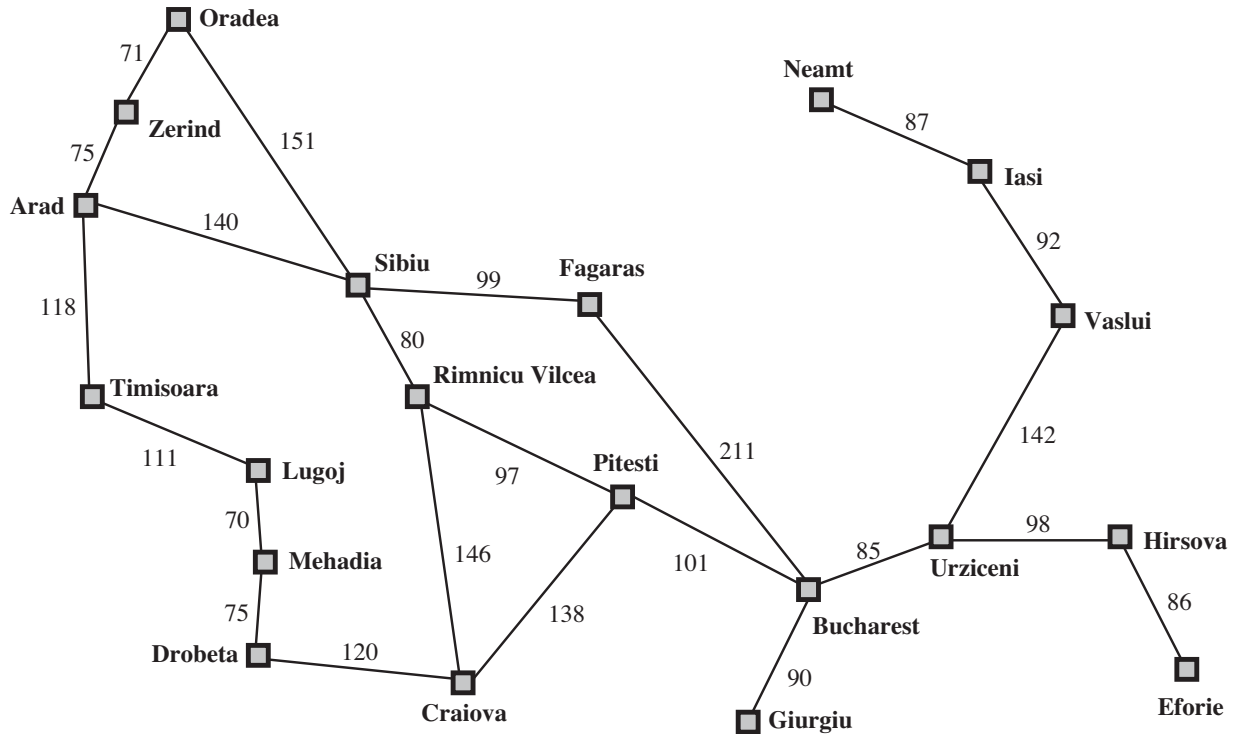Exercise 01.01 (3.3 AIMA3). Suppose two friends live in different cities on a map, such as the Romania map shown below.

On every turn, we can simultaneously move each friend to a neighbouring city on the map. The amount of time needed to move from city i to neighbour j is equal to the road distance $d(i, j)$ between the cities, but on each turn the friend that arrives first must wait until the other one arrives (and calls the first on his/her cell phone) before the next turn can begin.

We want the two friends to meet as quickly as possible.

a) Write a detailed formulation for this search problem. (You will find it helpful to define some formal notation here.)

b) Let $D(i, j)$ be the straight-line distance between cities i and j. Which of the following heuristic functions are admissible? i) $D(i, j)$; ii) $2D(i, j)$; iii) $D(i, j)/2$.

c) Are there completely connected maps for which no solution exists?

d) Are there maps in which all solutions require one friend to visit the same city twice? [*Hint*: Consider any of the solvable maps from c) and add a self-loop to any of the nodes.]

Exercise 01.02 (part of 3.15 AIMA3). Consider a state space where the start state is number 1 and each state k has two successors: numbers $2k$ and $2k + 1$.

a) Draw the portion of the state space for states 1 to 15;

b) Suppose the goal state is 11. List the order in which nodes will be visited for breadth-first search and depth-limited search with limit 3.

Exercise 01.03 (3.21 AIMA3). Prove each of the next statements, or give a counterexample:

a) Breadth-first search is a special case of uniform-cost search.

b) Depth-first search is a special case of best-first tree search.

c) Uniform-cost search is a special case of A* search.

Exercise 01.04 (3.23 AIMA3). Trace the operation of A^* search applied to the problem of getting from Oradea to Bucharest using the straight-line distance heuristic h_{SLD}.

Arad	366	**Mehadia**	241
Bucharest	0	**Neamt**	234
Craiova	160	**Oradea**	380
Drobeta	242	**Pitesti**	100
Eforie	161	**Rimnicu Vilcea**	193
Fagaras	176	**Sibiu**	253
Giurgiu	77	**Timisoara**	329
Hirsova	151	**Urziceni**	80
Iasi	226	**Vaslui**	199
Lugoj	244	**Zerind**	374

That is, show the sequence of nodes that the algorithm will consider and the f, g, and h score for each node.

Exercise 01.05 (3.29 AIMA3). Prove that if a heuristic is consistent, it must be admissible. Construct an admissible heuristic that is not consistent.