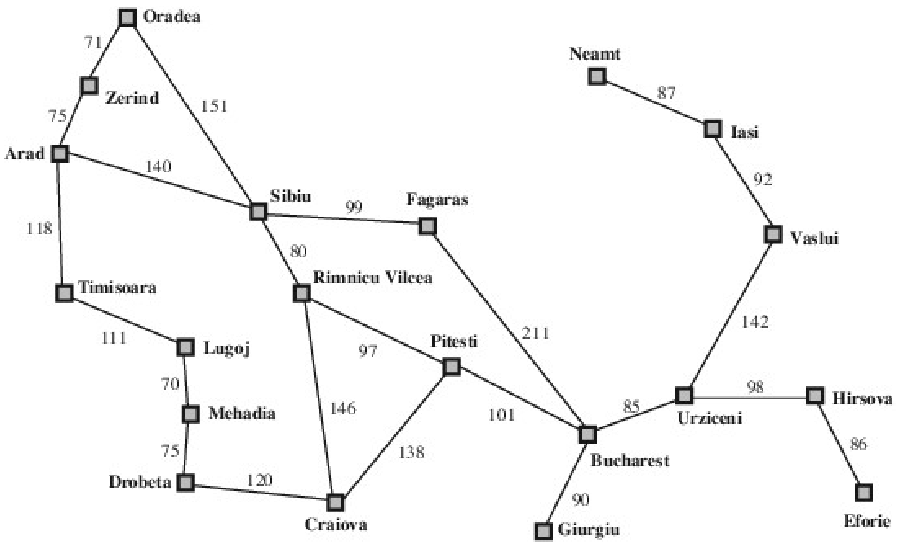
COSC1125/1127 Artificial Intelligence
School of Computer Science and IT
RMIT University
Semester 1, 2017

Tutorial Sheet 3
Heuristic and Adversarial Search

1. Consider the problem of getting from Arad to Bucharest in Romania and assume the straight line distance (SLD) heuristic will be used.



Straight−line distance to Bucharest	
Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

   (a) Give the part of the search space that is realized in memory and the order of node expansion for:
      i. Greedy search assuming that a list of states already visited is maintained.
      ii. A* search assuming that a list of states already visited is maintained.

   (b) How would the above searches differ if the list of states already visited is NOT maintained?

   (c) How do the above searches perform for planing a trip from Iasi to Fagaras? You do not need to do the detailed search (you are not given the heuristic function to Fagaras), but use the graphical map to extract the straight distance.

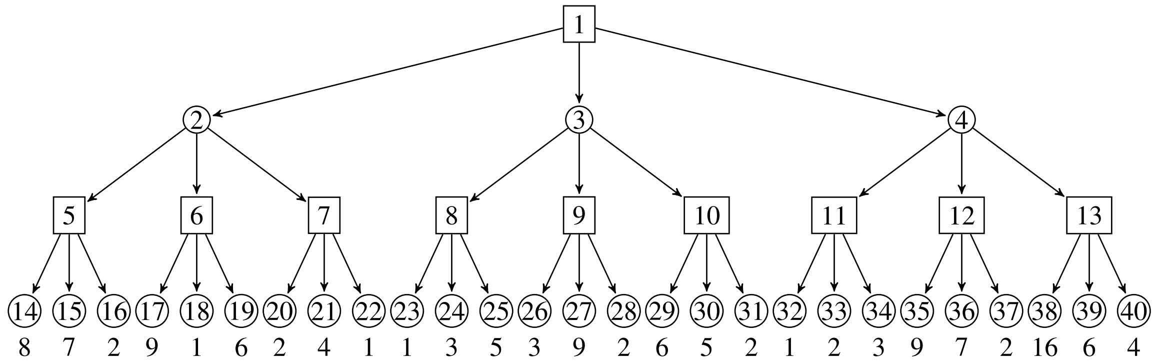
2. Suppose we run a greedy search algorithm with $h(n) = -g(n)$. What sort of search will result?

3. Suppose we run an A^* algorithm with $h(n) = 0$. What sort of search will result?

4. Explain why the set of states examined by A^* is a subset of those examined by breadth-first search when the cost of every step is always 1.

5. Is there a heuristic that would be useful for the missionaries and cannibals problem? The generalized missionaries and cannibals problem (n missionaries and n cannibals)?

6. Consider the following game tree. Player MAX plays first and is represented with rectangles; MIN player is represented with circles. Numbers in each node are names used for convenience to refer to them (starting node is node 1). Finally, utility of leaf nodes are shown below them (e.g., the utility of node 21 is 4).



(a) Use mini-max to determine the best move for MAX.

(b) Which nodes will not be examined if the alpha-beta procedure is used?

(c) In which order will the nodes be examined by the alpha-beta procedure (assuming its depth-first implementation)?

(d) Did the alpha-beta procedure give the same best move as mini-max?

7. Does A*'s search time always grow at least *exponentially* with the length of the optimal solution?

8. If $h(\cdot)$ is admissible and s is the start node, how is $h(s)$ related to the cost of the solution found by A* search?

9. Prove that if $h(n) = h^*(n)$ for all nodes n, then whenever A* expands a node x, x must lie on an optimal path to a goal.

10. Prove that if a heuristic is consistent, then it is also admissible. What about the converse?

11. Prove that A* without remembering nodes (i.e., without a closed list) is optimal when using an admissible heuristic.