EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

MATH. - NATURWISS. FAKULTÄT
FACHBEREICH INFORMATIK
KOGNITIVE SYSTEME · PROF. A. ZELL

# Artificial Intelligence

## Assignment 3

Assignment due by: 16.11.2016, Discussion: 22.11.2016

Tutors: Yann Berquin (yann.berquin@uni-tuebingen.de), Isabel Patiño (isabel.patino@uni-tuebingen.de) and Hauke Neitzel (hauke.neitzel@uni-tuebingen.de).

### Question 1 Greedy best-first search (6 points)

For this question use the tree search version of greedy best-first search.

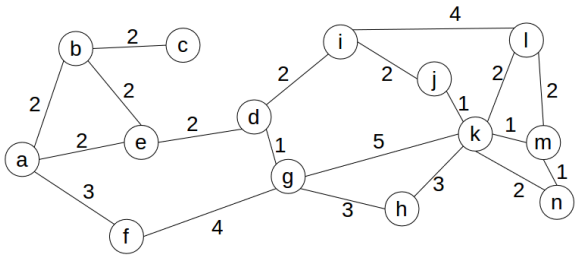



Table 1.

| $n$ | $h(n)$ | $n$ | $h(n)$ |
|---|---|---|---|
| $a$ | 15 | $h$ | 3 |
| $b$ | 14 | $i$ | 4 |
| $c$ | 10 | $j$ | 1 |
| $d$ | 7 | $k$ | 0 |
| $e$ | 11 | $l$ | 2 |
| $f$ | 12 | $m$ | 1 |
| $g$ | 5 | $n$ | 2 |

Table 2.

| $n$ | $h(n)$ | $n$ | $h(n)$ |
|---|---|---|---|
| $a$ | 17 | $h$ | 4 |
| $b$ | 16 | $i$ | 7 |
| $c$ | 12 | $j$ | 3 |
| $d$ | 9 | $k$ | 1 |
| $e$ | 13 | $l$ | 2 |
| $f$ | 14 | $m$ | 0 |
| $g$ | 8 | $n$ | 1 |

Table 3.

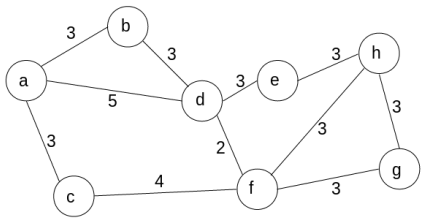
| $n$ | $h(n)$ | $n$ | $h(n)$ |
|---|---|---|---|
| $a$ | 17 | $h$ | 7 |
| $b$ | 15 | $i$ | 4 |
| $c$ | 11 | $j$ | 2 |
| $d$ | 9 | $k$ | 2 |
| $e$ | 13 | $l$ | 0 |
| $f$ | 15 | $m$ | 2 |
| $g$ | 9 | $n$ | 5 |

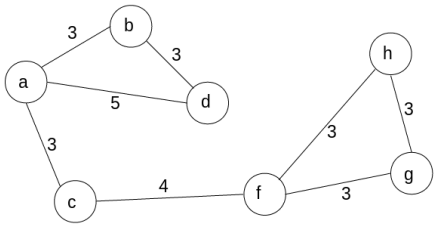
(a) Using the heuristic function shown in Table 1, is it possible to find a path between the node $\mathbf{a}$ and the target $\mathbf{k}$? Is the path optimal? Justify your answer.

(b) Using the heuristic function shown in Table 2, is it possible to find a path between the node $\mathbf{d}$ and the target $\mathbf{m}$? Is the path optimal? Justify your answer.

(c) Using the heuristic function shown in Table 3, is it possible to find the path from $\mathbf{b}$ to $\mathbf{l}$? Is the path optimal? What can you conclude about the Greedy best-first search algorithm? Justify your answer.

## Question 2 Pathfinding with A* (6 points)




| $n$ | $h(n)$ |
|---|---|
| $a$ | 10 |
| $b$ | 4 |
| $c$ | 7 |
| $d$ | 3 |
| $e$ | 0.5 |
| $f$ | 2 |
| $g$ | 1 |
| $h$ | 0 |

(a) Find the shortest path between the node $\mathbf{a}$ and the target node $\mathbf{h}$ using A* by hand for the left graph. Detail each intermediate step and indicate the f-values. Is there a unique shortest path?

(b) Using the right graph, try to find the shortest path between the node $\mathbf{a}$ and the target node $\mathbf{h}$ using A* by hand. Discuss your results.

(c) Construct a graph with five nodes, where A* changes the f-value for the target node at least three times. Can you make it change four times?

## Question 3 Programming in LISP (1+4+3 = 8 points)

Download the file *graphsearch-astar.lisp*, which contains a graph of German cities and the distances between them, as well as the coordinates of each city and some functions to access that information: `(expand city)` returns a list of all the cities connected to `city`, `(get-distance city1 city2)` returns the distance between two adjacent cities in km (or `nil` if they are not adjacent) and `(get-coordinates city)` returns the $xy$ position (in km) of `city` relative to a flat coordinate system. The file also contains function stubs for the different parts of this exercise.

(a) Implement an admissible heuristic function for A* route planning based on the geographical data available for the graph. ($h(n) = 0$ is not an acceptable answer).

(b) Implement A* graph search in LISP. The function you implement should return the total length of the path that was found, the path itself and a list of all the cities visited by the algorithm. You do not need to use efficient data-structures for priority-queues or sets when implementing this graph search algorithm (e.g. use simple sorted or unsorted list).

(c) Calculate the shortest path from any city to Hamburg and fill in a list of pairs $(c_i, distance(c_i, Hamburg))$ for all cities $c_i$, sorted in ascending order by distance. What would be the most efficient way to calculate this?