Einführung in Artificial Intelligence SS 2024, 4.0 VU, 192.027

Exercise Sheet 2 – Problem Solving and Search

For the discussion part of this exercise, mark and upload your solved exercises in **TUWEL** until Wednesday, June 5, 23:55 CEST. The registration for a solution discussion ends on Friday, June 7, 23:55 CEST. Be sure that you tick only those exercises that you can solve and explain!

In the discussion, students will be asked questions about their solutions of examples they checked. The discussion will be evaluated with 0–25 points, which are weighted with the fraction of checked examples and rounded to the next integer. There is *no minimum number of points* needed for a positive grade (i.e., you do not need to participate for a positive grade, but you can get at most ≈80% without doing exercises).
Note, however, that *your registration is binding*. Thus, *if* you register for a solution discussion, then it is *mandatory* to show up. No-show after a registration without plausible excuse leads to a penalty. Such students will not have the possibility to participate in another exam once they have gotten a certificate. If you registered but cannot show up due to unpredictable and unavoidable obstacles, either deregister or send us a confirmation (doctor's note, etc) and you will be excused. Please ask questions in the **TUWEL** forum or visit our tutors during the tutor hours (see **TUWEL**).

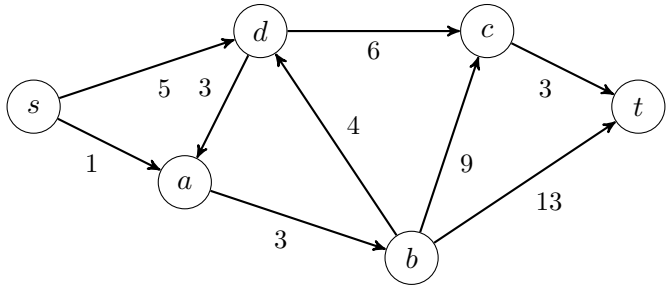
Exercise 2.1: Consider again the 8-Puzzle discussed in the lecture. Consider the discussed heuristics

(a) $h_1(n)$: number of misplaced tiles,

(b) $h_2(n)$: Manhattan distance.

Show whether the two suggested heuristics are admissible and/or consistent (monotonic).

Exercise 2.2: Perform the greedy algorithm and the A* algorithm using the given heuristic function h on the following graph in order to find a shortest path from s to t. In which order are the nodes expanded? Show the contents of the priority queue at each iteration. If multiple nodes have the same priority, expand the one that comes first alphabetically. Compare the shortest paths returned by the two algorithms, argue whether these solutions are optimal? Justify your answer.



$h(s) = 14, h(a) = 7, h(b) = 6, h(c) = 2, h(d) = 8, h(t) = 0$

Exercise 2.3: Assume h_a, h_b are admissible heuristics. Which of the following heuristics is admissible? Justify your answer.

(a) $h_1(n) = \frac{h_a(n)}{h_b(n)}$.

(b) $h_2(n) = 2^{(h_a+h_b)/c} - 1 \qquad c > 0$

Exercise 2.4: Let $f(n) = c \cdot g(n) + d \cdot h(n)$ be an evaluation function, where c and d are constants.

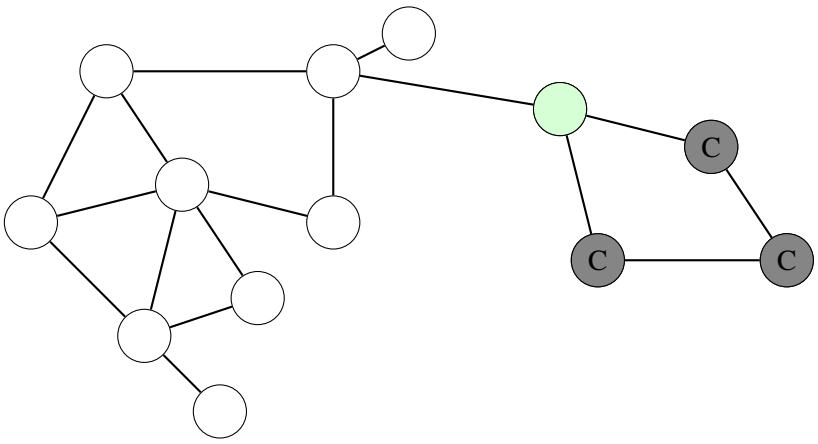
(1) Define c, d, $h(\cdot)$, $g(\cdot)$ such that A* with this evaluation function acts as a breadth-first search.

(2) Define c, d, $h(\cdot)$, $g(\cdot)$ such that A* with this evaluation function acts as a depth-first search.

(3) Define c, d, $h(\cdot)$, $g(\cdot)$ such that A* with this evaluation function acts as a uniform cost search.

You may assume that nodes contain all the information that we discussed in the lecture.

Exercise 2.5: Darth Vader has recently conquered a new sector of the galaxy. However, he now faces the problem that the Emperor has only provided him with 3 mobile command centers to secure the planets in this area. A planet is considered secure once either a command center is located on it or a neighboring planet has a command center. The hyperspace routes connecting the planets are depicted as edges in the following graph, with the planets as nodes. Darth Vader randomly distributed the command centers among the planets. Now it's your turn to execute a hill-climbing local search to find a goal state where each planet is covered. A step in the search process involves transferring one command center from one planet to another, the number of secured planets represents the value of each state. *Note: A command center can be transfered to any planet during a step, however only one can be transfered at a time.*



Covered Planets: 4

Exercise 2.6: Decide and explain which of the following statements are true and which are false? Back up your answers with proofs or counterexamples.

(1) Consider the basic implementation of local beam search as discussed in the lecture. A local search using this approach always outperforms *k* individual local searches running basic hill-climbing.

(2) Consider the basic implementation of the genetic algorithm as discussed in the lecture. To streamline the algorithm, one may choose to skip the mutation step at the end. This decision may prolong the computation time, but it is still possible for the algorithm to cover the entire search space.

Exercise 2.7: Consider the vacuum-cleaner world example discussed in the lecture. This time, a mischievous puppy is introduced into the world, presenting some challenges for our little robot friend. The following nondeterministic action effects have been incorporated into the base world:

(1) *Playmate.* The puppy sees the robot as its playmate, which introduces the possibility of our robot getting pushed around *after* it vacuums a space. For instance, the "Suck" action in the state [A, Dirt, Dirt] results in either [A, , Dirt] or [B, , Dirt]. *Note that the puppy pushes the robot, after it has already cleaned the space.*

(2) *Dog Hair.* As with most dogs, the puppy also sheds, which is why hair can get caught in the robot's tires if it moves without having vacuumed beforehand. Whenever the robot takes a "Move" action and moves away from a space that contains dirt, it is possible that the robot will spread the dirt on the new space. For example, the "Move-Left" action in the state [B, , Dirt] results in either [A, , Dirt] or [A, Dirt, Dirt].

Decide whether this search problem with nondeterministic action effects can be solved with an *And-Or Tree* or if a *Cyclic Solution* is required. Illustrate the corresponding solution visually.

Exercise 2.8: Perform the Online Search algorithm, as outlined in the lecture, on the following maze. Show the agent's location in the maze for each state along with the next move and the updates made to the *untried*, *result*, and *unbacktracked* arrays. Note that the preferred moves are in the order of Up (U), Right (R), Left (L), and Down (D).