SAARLAND UNIVERSITY ARTIFICIAL INTELLIGENCE
Dr. Álvaro Torralba, Prof. Wolfgang Wahlster
Dr. Cosmina Croitoru, Daniel Gnad, Marcel Steinmetz
Yannick Körber, Michael Barz
Christian Bohnenberger, Sören Bund-Becker, Sophian Guidara,
Alexander Rath, Khansa Rekik, Julia Wichlacz, Anna Wilhelm

Exercise Sheet 1.

Solutions due Tuesday, **May 8**, 16:00 – 16:15, in the lecture hall.[1]

Exercise 1. (1 Point)

Consider the state space depicted in Figure 1, where A is the initial state, and F and H are goal states. The transitions are annotated by their costs. List all states that are

(i) solvable,

(ii) dead-ends,

(iii) not reachable from B,

(iv) reachable from I.

[1]Solutions in paper form only, and solution submission only at the stated time at the stated place. At most 3 authors per solution. All authors must be in the same tutorial group. All sheets of your solution must be stapled together. At the top of the first sheet, you must write the names of the authors and the name of your tutor. Your solution must be placed into the correct box for your tutorial group. If you don't comply with these rules, 3 points will be subtracted from your score for this sheet.

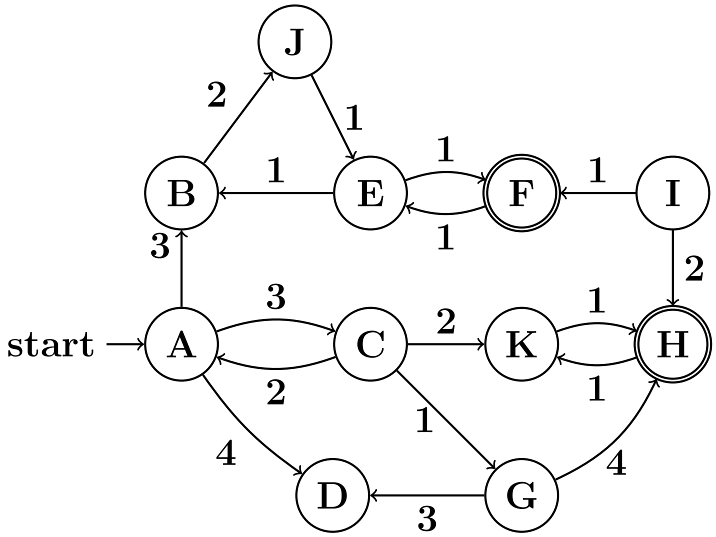

Figure 1: State space used throughout this sheet.

Exercise 2. (3 Points)

Consider again the state space depicted in Figure 1.

(a) Run uniform-cost search on this problem. Draw the search graph and annotate **each node** with its g **value and the order in which states are selected for expansion**. Draw duplicate nodes, and mark them accordingly by crossing them out. If the choice of the next state to be expanded is not unique, expand the lexicographically smallest state first. (eg. a before d :-)) Give the solution found by uniform-cost search. Is this solution guaranteed to be optimal? Justify your answer.

(b) Run iterative-deepening search until it finds a solution. For each depth depict the corresponding search tree. Annotate each state with the **order in which states are selected for expansion**. If the choice of the next state to be expanded is not unique, expand the lexicographically smallest state first. Give the solution found by iterative-deepening search. Is this solution guaranteed to be optimal? Justify your answer.

Exercise 3. (3.5 Points)

Consider again the state space from Figure 1.

(i) Run A^* search on this problem. As a heuristic estimate for a state s, use the minimal number of edges that are needed to reach a goal state from s (or ∞ if s is not solvable, e.g: h(C)=2). Draw the search graph and **annotate each node with the g and h value as well as the order of expansion**. Draw duplicate nodes as well, and mark them accordingly by crossing them out. If the choice of the next state to be expanded is not unique, expand the lexicographically smallest state first. Give the solution found by A^* search. Is this solution optimal? Justify your answer.

(ii) Run the hill climbing algorithm, as stated in the lecture, on this problem. **Use the heuristic function from part (i)**. For each state, provide all applicable actions and the states reachable using these actions. **Annotate states with their heuristic value. Specify which node is expanded in each iteration of the algorithm.** If the choice of the next state to be expanded is not unique, expand the lexicographically smallest state first. Does the algorithm find a solution? If yes, what is it and is it optimal?

(iii) Could hill-climbing stop in a local minimum without finding a solution? If yes, give such an example heuristic $h : \{A, B, ..., K\} \rightarrow \mathbb{N}_0^+ \cup \{\infty\}$ for the state space depicted in Figure 1, and explain what happens. If no please explain why.

Exercise 4. (2.5 Points)

Consider a variant of the Vacuum Cleaner problem from the lecture where a robot has to clean a 3×3 square room (see Figure 2). There are four possible actions: up, left, right, and down. There is no suck action since the robot automatically cleans any dirty spot it stands on. Hence, its task is moving around the room and visit all dirty spots. Throughout the exercise, we use Manhattan distance.

1. Which of these heuristics are admissible? Why / Why not? Give clear proof arguments. (Formal proofs are not needed.)

   Note: To prove a heuristic admissible, you need to show that it is admissible for an arbitrary input state of the illustrated 5×5 example. To show that a heuristic is not admissible, a counter example is sufficient.

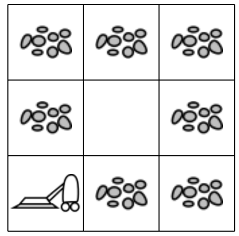

Figure 2: Illustration of the Vacuum Cleaner problem

- h_1 = Minimum distance from the robot to any dirty spot.
- h_2 = Maximum distance from the robot to any dirty spot.
- h_3 = Sum of the distances from the robot to all the dirty spots.
- h_4 = Number of moves needed to visit all spots.
- $h_5 = h_1 + h_2$.