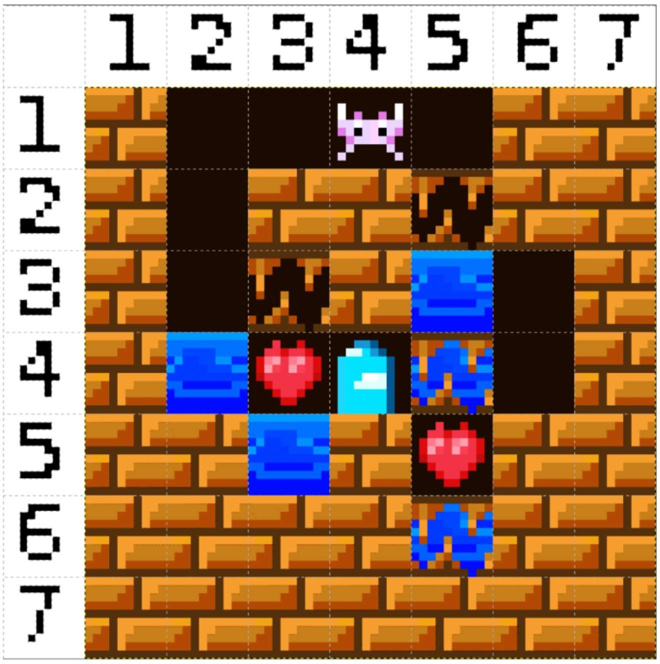
AI Exam

Section 1



AIMaL is a planet made of blocks. The figure to the left depicts a fragment of that planet. Our outer-world hero and agent (Unit One) stands in the middle of the top side, ready for another new adventure. It wants to reach the exit (the door) by traversing the perilous paths that lead it there. Unit One has four actions at its disposal: move right (R), move left (L), move up (U), move down (D). Moving through dark blocks requires 5 fatigue points. Dodging stalagmites and stalactites requires 20 fatigue points. Swimming in the water causes an expenditure of 30 fatigue points. 50 fatigue points are necessary when these two elements are combined in a single block. The hearts along the way restore our hero, who needs just 1 fatigue point to pass through that block. Going through the exit door implies a fatigue quantifiable in 1 point, too. A wall of bricks impedes Unit One from moving forward. Coordinates are marked on the picture frame.

Considering the above as a search problem, let us use the following scheme:

- Every traversable block (so, not the ones with bricks) is a state Unit One can reach;
- The initial state is the one in which Unit One stands in the picture above (coordinates:. 4,1);
- The goal state is where the exit door lies (coordinates: 4,4);
- Actions are R, L, U, D (see above);
- The transition model is determined by the moves of Unit One;
- The cost of the actions is measured in fatigue points.

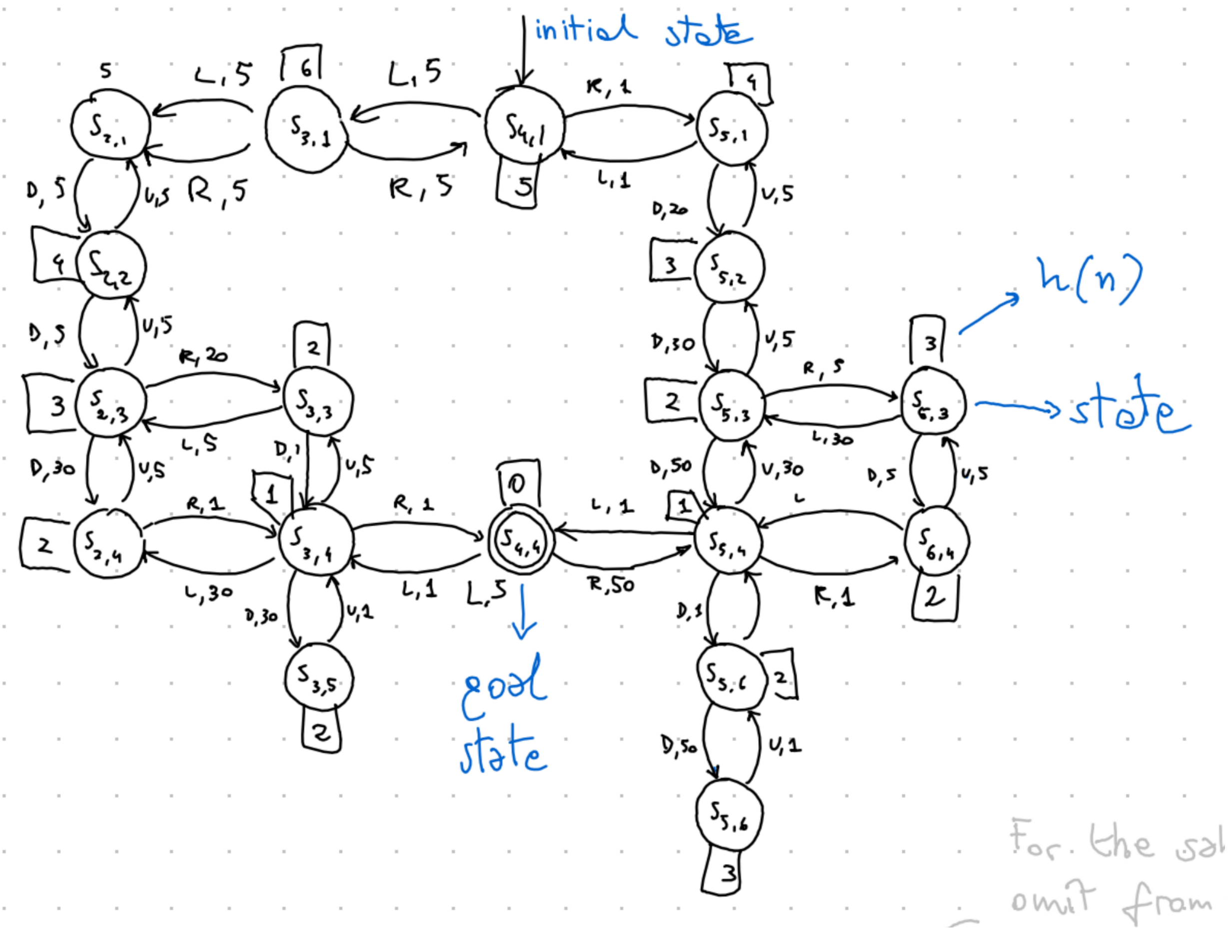
ⓘ Required notions

- Heuristics
- A*
- Breadth First Search
- Depth First Search(possibly)
- Best First Search(possibly)

1. [1 pt.] Represent the state space as a graph. Use the (x, y) coordinates to identify the states. For example, Unit 1 begins its adventures in $s_{4,1}$, looking forward to reach the exit door in $s_{4,4}$. Decorate the state space graph with a heuristic $h(n)$ defined as follows: $h(n)$ is the distance in blocks from n to the exit, computed as the number of moves Unit One would need to reach the exit door via the shortest path with the uniform cost of 1 (for example, in the initial state it is equal to 5).

Draw the state space as a graph

Here we just need to basically draw an automaton with edges that have a cost.
The cost will be the heuristic $h(n)$ which is just the shortest number of blocks that can be traveled to reach the exit.



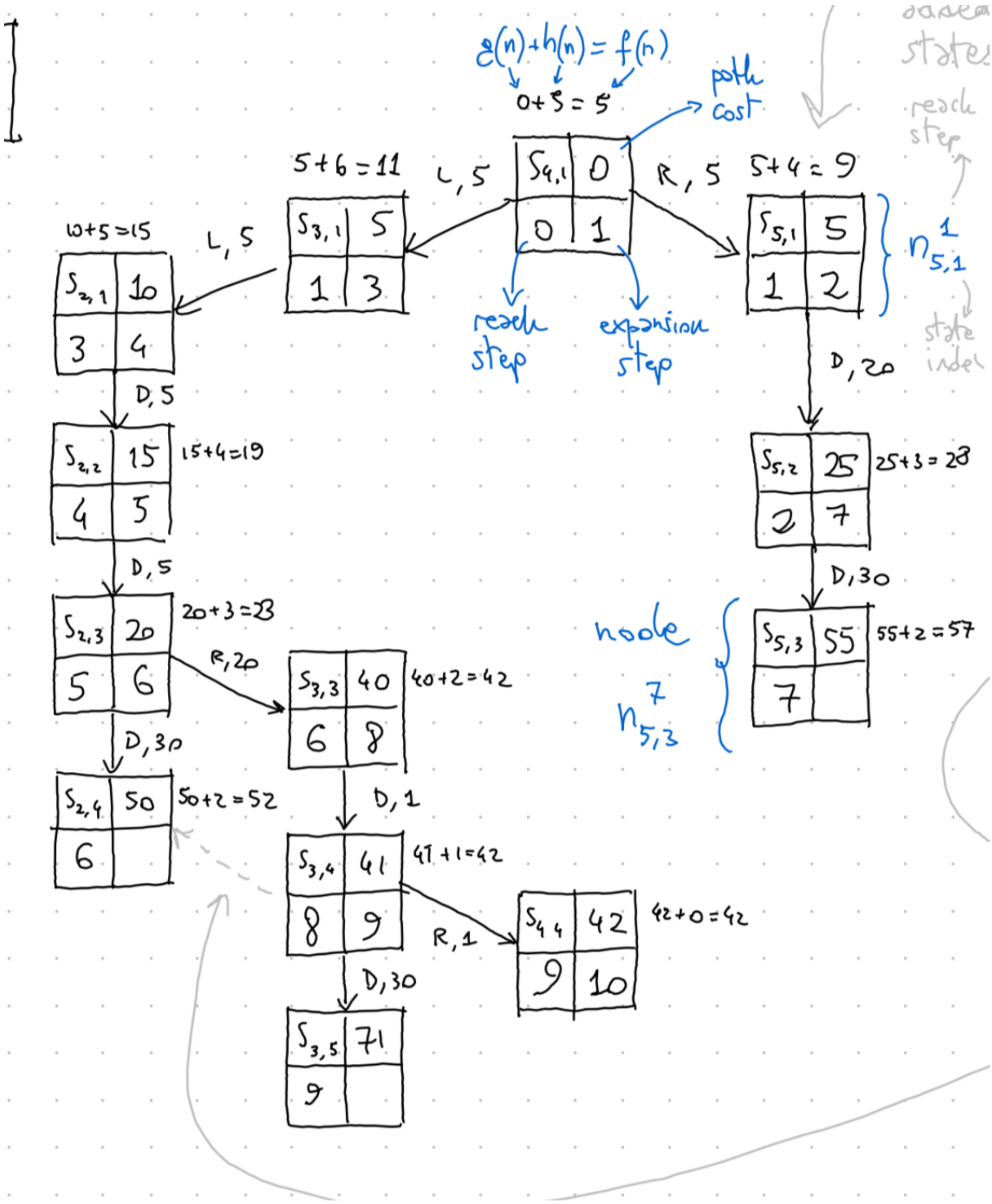
Here, in addition to the cost, the prof has included the direction that the edge takes us to.

---

2. [2 pts.] Using the heuristic above, run the A^* algorithm to find a solution. Make sure to describe the procedure, show the search tree specifying the order in which nodes are reached and expanded, and indicate the found path with its cost. In case the evaluation function for two nodes returns the same value, prioritise the node associated with the state with the highest x index, and (in case of another tie) with the highest y index.

Run A*

We just add to the "reached" set the node in the "frontier" that has the lowest cost.



In the squares above:

- Top-left: state
- Top-right: actual cost to reach
- Bottom-left: step at which they are added to "frontier"
- Bottom-right: step in which they are added to "reached"

	frontier	reached (updates)
0	$\langle n_{4,1} \rangle$	$s_{4,1} \mapsto n^0_{4,1}$
1	$\langle n^1_{5,1}, n^1_{3,1} \rangle$	$s_{5,1} \mapsto n^1_{5,1}$, $s_{3,1} \mapsto n^1_{3,1}$
2	$\langle n^1_{3,1}, n^2_{5,2} \rangle$	$s_{5,2} \mapsto n^2_{5,2}$
3	$\langle n^3_{2,1}, n^2_{5,2} \rangle$	$s_{2,1} \mapsto n^3_{2,1}$
4	$\langle n^4_{2,2}, n^2_{5,2} \rangle$	$s_{2,2} \mapsto n^4_{2,2}$
5	$\langle n^5_{2,3}, n^2_{5,2} \rangle$	$s_{2,3} \mapsto n^5_{2,3}$
6	$\langle n^2_{5,2}, n^6, n^6_{2,4} \rangle$	$s_{3,3} \mapsto n^6_{3,3}$, $s_{2,4} \mapsto n^6_{2,4}$
7	$\langle n^6_{3,3}, n^6_{2,4}, n^7_{5,3} \rangle$	$s_{5,3} \mapsto n^7_{5,3}$
8	$\langle n^8_{3,4}, n^6_{2,4}, n^7_{5,3} \rangle$	$s_{3,4} \mapsto n^8_{3,4}$
9	$\langle n^9_{4,4}, n^6_{2,4}, n^7_{5,3}, n^9_{3,5} \rangle$	$s_{4,4} \mapsto n^9_{4,4}$, $s_{3,5} \mapsto n^9_{3,5}$
10	$\langle n^6_{2,4}, n^7_{5,3}, n^9_{3,5} \rangle$	

3. [1 pt.] Indicate whether the breadth-first search algorithm considering a uniform cost of 1 per move can be used to compute an admissible heuristic (and if so, explain how) or not (and in case, clarify why).

Can BFS constitute an admissible heuristic

A heuristic function is said to be admissible if it never overestimates the cost of reaching the goal. Also it must have a value for every input.

WIP

Section 2

Section 3

Required Notions

- Propositional Logic
- Satisfiability of a Compound Proposition
- Contrapositive of a proposition
- Forward Chaining
- Entailment of propositions
- Tableau Resolution Deductive System
- DPLL

1. [1 pt.] Use the standard DPLL procedure with the max-degree heuristic to indicate whether the following formula is satisfiable. If it is, provide an interpretation that satisfies it. Please clarify every step taken by the procedure.

$$\gamma \doteq \big(a \vee \neg(\neg e \wedge \neg c)\big) \wedge \big(c \rightarrow (a \vee b)\big) \wedge \neg\neg(c \vee d) \wedge (e \rightarrow \neg a) \wedge (e \rightarrow b)$$