Intro to AI  
HW2

Diar Bathish - 213859325 - Diar.bathish@campus.technion.ac.il  
Moneer Khoury - 206443707 - Munir.khoury@campus.technion.ac.il

Part A:

In short, the state space is a 5x5 matrix with the positions of the robots, their remaining battery and their credit, the position of charging stations, packages, and their destinations.

The actions that each one of the robots can do.

The transition function.

The cost function, we pay one charge for each transition, and when we want to recharge our battery, we pay our entire credit and convert all of it to battery charge.

The initial state, it’s a random state in the state space.

The reward function, if we deliver a package to its destination, we receive a reward to our credit the equivalent of the Manhattan distance from the original position of the package to its destination, multiplied by 2.

Part B:

1. Advantages: the simple heuristic is faster and easier to calculate than the more complicated one, which allows us to dive deeper into the tree.

Disadvantages: the simple heuristic is less informed about the environment of the problem, and thus it’s less accurate than the complicated one.

1. She is wrong.

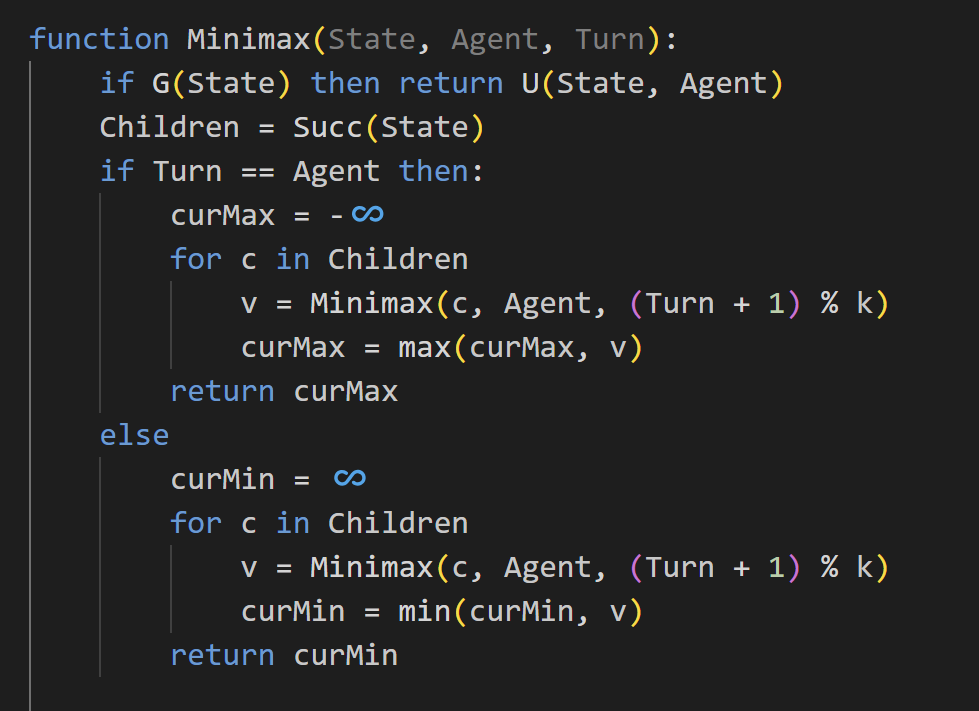
the algorithm does in fact return an optimal solution, but it doesn’t return the shortest path because it doesn’t take into consideration the number of steps it had already done during the search.

דוגמה נגדית

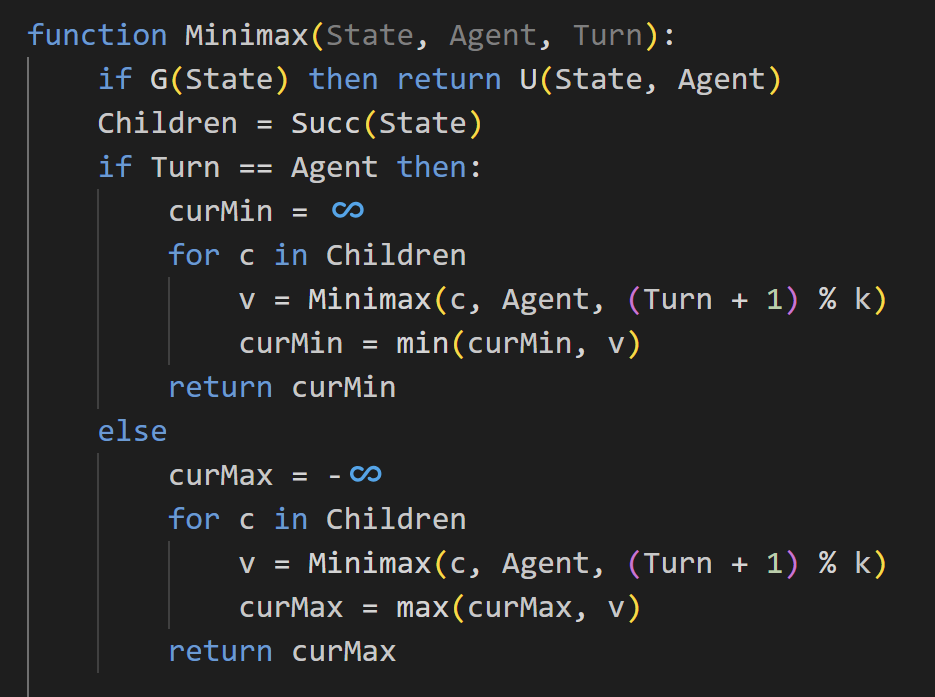
1. We will use a timer that tracks the time that has already passed and iteratively run the algorithm, each time increasing the max depth until the time ends, which in that case we will stop and return the current best result.

The group of algorithms is called , one example of is .

1. The world has now agents instead of .
2. Each agent has one goal in mind, to win, and it doesn’t care about other agents’ utility.



1. Each agent has one goal in mind, to minimize our utility.



1. Each agent has one goal in mind, that the agent coming after it has a maximum utility.

A computer screen shot of a code

Description automatically generated

Part C:

1. Yes, does in fact behave differently than .

In the sense of runtime it perform a bit better, in other words it’s a bit faster, but in the sense of deciding actions, they are different because is more informed due to the fact that it has more time (it doesn’t waste its entire on traversing children that we won’t go into because we already have a better pick than them), thus it can traverse more children in the same given time as , and thus it has more information about the world and has more information about what is the best move to do.

**But,** if there were no time limit, then both agents would behave the same since at the end the would ‘catch up’ with the , and will take the same decisions.

**And** if the children were exactly ordered in a way such that the first child of a has a the minimum value and the last one has the maximum value, and for a the first child has the maximum value and the last child has the minimum value, in this case, the would have to open the entire tree, and it can’t make any pruning because it always finds a better value than it previously had, so in this case both algorithms have a similar runtime.

Part D:

1. In case of a random enemy, we would prefer to use a Uniform probability, because the agent (enemy) picks his actions in a random way which means it uses a uniform probability to pick a specific action.
2. Sinse the max value of the heuristic is , thus if the agent finds a value ( is a child of ), then the agent can simply prune the remaining children of because it won’t find a higher value then .