**Task 1**

**Implement a basic driving agent**

**Question: what you see in the agent’s behavior. Does it eventually make it to the target location?**

The agent randomly chooses an action from “stay put”, “turn left”, “turn right” or “go forward” at each time point. The agent’s action is in disagree with the one that the planner told it (as shown right next to the car). When the agent’s action is “forward”, it knows to wait for the traffic light to turn green before it makes a move. When the agent’s action is “left”, it knows to wait on red. When the light turns green, it knows to yield for the oncoming cars. When the agent’s action is “right”, if the light is green, it will turn right right away. If the light is red, it will look at its left, if there is no car on its left or if the car on its left doesn’t want to go straight, it will turn right.

After a lot of driving around, wandering back and forth, the agent can eventually reach the destination.

**Task 2**

**Identify and update sate**

**Identify a set of states that are appropriate for modeling the driving agent. Justify why you picked these set of states, and how they model the agent and its environment.**

I pick the set of states as 8 variables and put them in a namedtuple data structure.

1. Location: this is the point where the agent is currently at, (col, row).
2. Heading: this is the direction of the agent is facing,
   1. (1, 0) east
   2. (0, -1) north
   3. (-1, 0) west
   4. (0, 1) south
3. Destination: this is the target point where the agent is going to, when the agent reaches this point, the game is done.
4. Deadline: this number indicates how many more steps can agent go before gam eover.
5. Light: red or green. This is the traffic light for agent.
6. Oncoming: this is to indicate if there is another car coming from the front at the same intersection with the agent. If the value is “None”, it means there is no car coming from front. Otherwise, its value indicates the direction the other car is going, e.g. “left”, “right”, “forward”.
7. Left: this is to indicate if there is another car coming from the left at the same intersection with the agent.
8. Right: this is to indicate if there is another car coming from the right at the same intersection with the agent.

I picked these variables because they are most relevant for agent to make the optimal decisions. Variable 1 and 3 keeps track of the current position and destination position. This is important because the agent needs to know where it is and where it wants to go to make the right decision. Variable 2, the heading of the agent is important because, firstly, it determines which cars are on its left, right or front, and the next moves of those cars would potentially impact how the agent reacts; secondly, the direction which the agent is facing also impacts the next\_waypoint value even if the agent is going to move between two same points. Variable 4, the deadline is important because the optimal solution might be different based on the remaining time. For example, if there are a lot of steps left, the agent would be willing to delay the rewards; it would like to take some small rewards in order to get a larger reward later. However, if there are only a few steps left, the agent would be more short-sighted and greedy, its priority would be to collect immediate rewards as much as it can. Variable 5, 6,7, 8 are the information for the surroundings. The light status is important because the agent needs to obey the traffic rules and not move or yield to traffic when the light is red. The information of other cars that are at the same intersection is important because our agent need to know if it should yield to other traffics. If it runs into other cars, it should get a big penalty.