Machine Learning Engineer Nanodegree

Smartcab

In your report, mention what you see in the agent’s behavior. Does it eventually make it to the target location?

Random agent behavior has been implemented by following code:

```
# TODO: Update state

self.state = inputs['location']

self.heading=inputs['heading']

action = random.choice(Environment.valid_actions)

reward = self.env.act(self, action)
```

Agent is moving randomly. Rules of the environment (traffic light, other traffic) are not respected. After few attempts (10) the agent has not reached destination before the deadline in any of attempts. Actions of the agent are not in line with planner.

Justify why you picked these set of states, and how they model the agent and its environment.

As a set of states I’ve picked the following parameters:

- self.next_waypoint: next action suggested by simple planner
- Inputs[**'light'**]: traffic light
- Inputs[**'oncoming'**]: oncoming traffic
- Inputs[**'left'**]: traffic from the left

First parameter allows me to find the direction toward the goal. 3 other parameters give the opportunity to pick actions available in this environment. So on each timestep I have understanding where is my goal and which actions are permitted. Traffic from the right is not needed to describe the environment, because according to the rules in the environment it is not impacting our policy (DummyAgent() update() function).

Independently of traffic from the right the agent could select actions allowed in the environment. In case of deadline included, the dimension of the state space will increase => convergence time will of Q values will increase.

What changes do you notice in the agent's behavior?

The agent starts moving towards the goal and reaching the goal before the deadline.

Report what changes you made to your basic implementation of Q-Learning to achieve the final version of the agent. How well does it perform?

I made the following adjustments:

1) In order to eliminate cycles in car movements on the initial step, I've add 0.1 reward for planner recommendation action from the beginning.

```
self.Q[self.state][self.state[0]]=self.Q[self.state][self.state[0]]+0.1
```

2) Epsilon-greedy exploration vs exploitation strategy has been implemented . On the early time in first runs initial agent actions could be chosen randomly for better exploration of environment, but later Qmax policy prevails.

```
self.epsilon=1/float(t+self.current_trial+1)
if random.random() < self.epsilon:
  proposed_action = random.choice(self.actions)
  maxQ = self.Q[self.state][proposed_action]
else:
  maxQ = max(self.Q[self.state][next_action] for next_action in self.actions)
```

No hard coding has been used.

Does your agent get close to finding an optimal policy, i.e. reach the destination in the minimum possible time, and not incur any penalties?

Statistics to benchmark the agent behavior based on (success_rate, penalty_ratio) has been implemented.

Following alpha/gamma combinations has been tested and compared based on 10 trials:

	Alpha=0,1	Alpha=0,9
Gamma=0,1	('10/10 = %100.0', '53/143 = %37.06')	('9/10 = %90.0', '13/213 = %6.1')
Gamma=0,9	('10/10 = %100.0', '55/144 = %38.19')	('10/10 = %100.0', '12/160 = %7.5')

From the table above, we can conclude that parameter Alpha is controlling how strictly the agent is following traffic rules. In the case of Gamma=0.1 and Alpha=0.9 in the first trial the agent has not reached the destination in the time limit. So assuming our tolerance to the violation of rules during the learning period, the optimal behavior of the agent is reaching with Alpha=0.9 and Gamma=0.9.

In order to investigate the agent behavior after education simulation with n_trials=100 has been launched (self.alpha=0.9, self.gamma=0.9). Log of last 10 trials is provided in Appendix 1.

For how many (latest) trials were the agent able to run without incurring penalty?

During last 10 trials the score has changed from ('86/90 = %95.6', '41/1332 = %3.08') to ('94/99 = %94.9', '46/1478 = %3.11'). It is mean that 5 penalties has been received.

Did the agent reach destination within allotted time first or following traffic rules correctly first?

After 100 trials the results are following: ('95/100 = %95', '47/1491 = %3.15'). It is meant that agent reached destination prior to deadline in 95 trials out of 100 and has 47 violations of traffic rules.

During the last 10 trials the agent is behaving optimally during the majority of actions: is is reaching the destination before the deadline, there is almost no violations of traffic rules. For example in the last trial:

State	**Action**	**Reward**

{next_waypoint:’forward’,'left': None, 'light': 'green', 'oncoming': None}	forward	2
{next_waypoint:’right’, 'light': 'green', 'oncoming': None, 'left': None}	right	2
{next_waypoint:’forward’, 'light': 'green', 'oncoming': None,'left': None}	forward	12

So agent getting positive rewards, selecting the action based on max Q and reaching the destination before the deadline.

Appendix 1

```
Simulator.run(): Trial 90

Environment.reset(): Trial set up with start = (1, 5), destination = (7, 2), deadline = 45

RoutePlanner.route_to(): destination = (7, 2)

('86/90 = %95.6', '41/1332 = %3.08')

LearningAgent.update(): deadline = 45, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (1, 5), 'heading': (-1, 0), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 44, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (1, 4), 'heading': (0, -1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 43, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (2, 4), 'heading': (1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 42, inputs = {'right': 'left', 'light': 'green', 'oncoming': None, 'location': (3, 4), 'heading': (1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 41, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 4), 'heading': (1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 40, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 4), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 39, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 4), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 38, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 4), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 37, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 4), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 36, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 4), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 35, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 4), 'heading': (1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 34, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 4), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 33, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 4), 'heading': (1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 32, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (7, 4), 'heading': (1, 0), 'left': None}, action = left, reward = 2

LearningAgent.update(): deadline = 31, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (7, 3), 'heading': (0, -1), 'left': None}, action = None, reward = 1

Environment.act(): Primary agent has reached destination!

LearningAgent.update(): deadline = 30, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (7, 3), 'heading': (0, -1), 'left': None}, action = forward, reward = 12

Simulator.run(): Trial 91

Environment.reset(): Trial set up with start = (7, 4), destination = (5, 1), deadline = 25
```

```
RoutePlanner.route_to(): destination = (5, 1)

('87/91 = %95.6', '41/1348 = %3.04')

LearningAgent.update(): deadline = 25, inputs = {'right': None, 'light': 'red', 'oncoming': 'forward', 'location': (7, 4), 'heading': (-1, 0), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = 24, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (7, 3), 'heading': (0, -1), 'left': None}, action = left, reward = 2

LearningAgent.update(): deadline = 23, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 3), 'heading': (-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 22, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 3), 'heading': (-1, 0), 'left': None}, action = right, reward = 2

Environment.act(): Primary agent has reached destination!

LearningAgent.update(): deadline = 21, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 2), 'heading': (0, -1), 'left': None}, action = forward, reward = 12

Simulator.run(): Trial 92

Environment.reset(): Trial set up with start = (6, 4), destination = (2, 5), deadline = 25

RoutePlanner.route_to(): destination = (2, 5)

('88/92 = %95.7', '41/1353 = %3.03')

LearningAgent.update(): deadline = 25, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 4), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 24, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 4), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 23, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 4), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 22, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 4), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 21, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 4), 'heading': (-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 20, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 4), 'heading': (-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 19, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 4), 'heading': (-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 18, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (3, 4), 'heading': (-1, 0), 'left': None}, action = forward, reward = 2

Environment.act(): Primary agent has reached destination!

LearningAgent.update(): deadline = 17, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (2, 4), 'heading': (-1, 0), 'left': None}, action = left, reward = 12

Simulator.run(): Trial 93

Environment.reset(): Trial set up with start = (8, 5), destination = (2, 6), deadline = 35

RoutePlanner.route_to(): destination = (2, 6)

('89/93 = %95.7', '41/1362 = %3.01')
```

```
LearningAgent.update(): deadline = 35, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (8, 5), 'heading': (0,
1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 34, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (7, 5), 'heading':
(-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 33, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 5), 'heading':
(-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 32, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 5), 'heading': (-1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 31, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 5), 'heading':
(-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 30, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (4, 5), 'heading': (-1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 29, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (4, 5), 'heading': (-1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 28, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (4, 5), 'heading': (-1,
0), 'left': None}, action = forward, reward = -1

LearningAgent.update(): deadline = 27, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (4, 5), 'heading': (-1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 26, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (4, 5), 'heading': (-1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 25, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 5), 'heading':
(-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 24, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (3, 5), 'heading':
(-1, 0), 'left': None}, action = forward, reward = 2

Environment.act(): Primary agent has reached destination!

LearningAgent.update(): deadline = 23, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (2, 5), 'heading':
(-1, 0), 'left': None}, action = left, reward = 12

Simulator.run(): Trial 94

Environment.reset(): Trial set up with start = (6, 5), destination = (8, 2), deadline = 25

RoutePlanner.route_to(): destination = (8, 2)

('90/94 = %95.7', '42/1375 = %3.05')

LearningAgent.update(): deadline = 25, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 5), 'heading': (0,
-1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 24, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (7, 5), 'heading':
(1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 23, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (8, 5), 'heading': (1,
0), 'left': None}, action = left, reward = -1

LearningAgent.update(): deadline = 22, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (8, 5), 'heading': (1,
0), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = 21, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (8, 6), 'heading':
(0, 1), 'left': None}, action = right, reward = 2
```

```
LearningAgent.update(): deadline = 20, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (7, 6), 'heading': (-1,
0), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 19, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (7, 5), 'heading':
(0, -1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 18, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (8, 5), 'heading':
(1, 0), 'left': None}, action = left, reward = 2

LearningAgent.update(): deadline = 17, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (8, 4), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 16, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (8, 4), 'heading':
(0, -1), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 15, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (8, 3), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 14, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (8, 3), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 13, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (8, 3), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 12, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (8, 3), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 11, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (8, 3), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 10, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (8, 3), 'heading':
(0, -1), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = 9, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (1, 3), 'heading': (1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 8, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (1, 3), 'heading': (1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 7, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (1, 3), 'heading': (1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 6, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (1, 3), 'heading': (1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 5, inputs = {'right': None, 'light': 'green', 'oncoming': 'left', 'location': (1, 3), 'heading': (1,
0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 4, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 3), 'heading': (1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 3, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 3), 'heading': (1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 2, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 3), 'heading': (1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 1, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 3), 'heading': (1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 0, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (2, 3), 'heading': (1,
0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = -1, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (3, 3), 'heading':
(1, 0), 'left': None}, action = forward, reward = 2
```

```
LearningAgent.update(): deadline = -2, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (4, 3), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = -3, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 3), 'heading': (1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = -4, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (1, 0), 'left': None}, action = forward, reward = -1

LearningAgent.update(): deadline = -5, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = -6, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = -7, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 3), 'heading': (1, 0), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = -8, inputs = {'right': 'forward', 'light': 'red', 'oncoming': None, 'location': (5, 4), 'heading': (0, 1), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = -9, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 4), 'heading': (-1, 0), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = -10, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 3), 'heading': (0, -1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = -11, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = -12, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = -13, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = -14, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = -15, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 3), 'heading': (1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = -16, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 3), 'heading': (1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = -17, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (7, 3), 'heading': (1, 0), 'left': None}, action = forward, reward = 2

Environment.act(): Primary agent has reached destination!

LearningAgent.update(): deadline = -18, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (8, 3), 'heading': (1, 0), 'left': None}, action = left, reward = 2

Simulator.run(): Trial 95

Environment.reset(): Trial set up with start = (6, 1), destination = (5, 5), deadline = 25

RoutePlanner.route_to(): destination = (5, 5)

('90/95 = %94.7', '44/1419 = %3.1')

LearningAgent.update(): deadline = 25, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 1), 'heading': (1, 0), 'left': None}, action = right, reward = 2
LearningAgent.update(): deadline = 24, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 2), 'heading': (0, 1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 23, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 2), 'heading': (-1, 0), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = 22, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 1), 'heading': (0, -1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 21, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 1), 'heading': (1, 0), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 20, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 2), 'heading': (0, 1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 19, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 2), 'heading': (-1, 0), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = 18, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 1), 'heading': (0, -1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 17, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 1), 'heading': (1, 0), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 16, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 2), 'heading': (0, 1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 15, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 2), 'heading': (-1, 0), 'left': None}, action = left, reward = 2

LearningAgent.update(): deadline = 14, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (0, 1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 13, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 3), 'heading': (0, 1), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 12, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 4), 'heading': (0, 1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 11, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 4), 'heading': (0, 1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 10, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 4), 'heading': (0, 1), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = 9, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (4, 4), 'heading': (-1, 0), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 8, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 3), 'heading': (0, -1), 'left': 'left'}, action = right, reward = 2

LearningAgent.update(): deadline = 7, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 3), 'heading': (1, 0), 'left': None}, action = right, reward = 2

Environment.act(): Primary agent has reached destination!

LearningAgent.update(): deadline = 6, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 4), 'heading': (0, 1), 'left': None}, action = forward, reward = 12

Simulator.run(): Trial 96

Environment.reset(): Trial set up with start = (6, 2), destination = (3, 4), deadline = 25

RoutePlanner.route_to(): destination = (3, 4)
```

```
('91/96 = %94.8', '44/1439 = %3.06')

LearningAgent.update(): deadline = 25, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 2), 'heading': (0, -1), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = 24, inputs = {'right': 'right', 'light': 'green', 'oncoming': None, 'location': (7, 2), 'heading': (1, 0), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 23, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (7, 3), 'heading': (0, 1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 22, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 3), 'heading': (-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 21, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 20, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 19, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 18, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 3), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 17, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 3), 'heading': (-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 16, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (4, 3), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 15, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (4, 3), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 14, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (4, 3), 'heading': (-1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 13, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 3), 'heading': (-1, 0), 'left': None}, action = forward, reward = 2

Environment.act(): Primary agent has reached destination!

LearningAgent.update(): deadline = 12, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (3, 3), 'heading': (-1, 0), 'left': None}, action = left, reward = 12

Simulator.run(): Trial 97

Environment.reset(): Trial set up with start = (1, 5), destination = (7, 5), deadline = 30

RoutePlanner.route_to(): destination = (7, 5)

('92/97 = %94.8', '44/1453 = %3.03')

LearningAgent.update(): deadline = 30, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (1, 5), 'heading': (0, -1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 29, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (2, 5), 'heading': (1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 28, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (3, 5), 'heading': (1, 0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 27, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (3, 5), 'heading': (1, 0), 'left': None}, action = forward, reward = -1
LearningAgent.update(): deadline = 26, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (3, 5), 'heading': (1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 25, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (3, 5), 'heading':
(1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 24, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 5), 'heading':
(1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 23, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 5), 'heading':
(1, 0), 'left': None}, action = forward, reward = 2

Environment.act(): Primary agent has reached destination!

LearningAgent.update(): deadline = 22, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 5), 'heading':
(1, 0), 'left': None}, action = forward, reward = 12

Simulator.run(): Trial 98

Environment.reset(): Trial set up with start = (4, 6), destination = (2, 3), deadline = 25

RoutePlanner.route_to(): destination = (2, 3)

('93/98 = %94.9', '45/1462 = %3.08')

LearningAgent.update(): deadline = 25, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 6), 'heading':
(0, 1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 24, inputs = {'right': None, 'light': 'red', 'oncoming': 'forward', 'location': (3, 6), 'heading':
(-1, 0), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = 23, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (3, 5), 'heading': (0,
-1), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = 22, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 5), 'heading':
(1, 0), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 21, inputs = {'right': 'forward', 'light': 'green', 'oncoming': None, 'location': (4, 6),
'heading': (0, 1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 20, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (3, 6), 'heading':
(-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 19, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 6), 'heading': (-1,
0), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 18, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 5), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 17, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 5), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 16, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (2, 5), 'heading':
(0, -1), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 15, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 4), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 14, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 4), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 13, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 4), 'heading': (0,
-1), 'left': None}, action = None, reward = 1
```

```
LearningAgent.update(): deadline = 12, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 4), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 11, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (2, 4), 'heading': (0,
-1), 'left': None}, action = None, reward = 1

Environment.act(): Primary agent has reached destination!

LearningAgent.update(): deadline = 10, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (2, 4), 'heading':
(0, -1), 'left': None}, action = forward, reward = 12

Simulator.run(): Trial 99

Environment.reset(): Trial set up with start = (6, 6), destination = (3, 4), deadline = 25

RoutePlanner.route_to(): destination = (3, 4)

('94/99 = %94.9', '46/1478 = %3.11')

LearningAgent.update(): deadline = 25, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 6), 'heading': (0,
-1), 'left': None}, action = left, reward = -1

LearningAgent.update(): deadline = 24, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 6), 'heading': (0,
-1), 'left': None}, action = right, reward = 0.5

LearningAgent.update(): deadline = 23, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (7, 6), 'heading': (1,
0), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 22, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (7, 1), 'heading':
(0, 1), 'left': None}, action = right, reward = 2

LearningAgent.update(): deadline = 21, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 1), 'heading': (-1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 20, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (6, 1), 'heading': (-1,
0), 'left': 'left'}, action = right, reward = 0.5

LearningAgent.update(): deadline = 19, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (6, 6), 'heading':
(0, -1), 'left': None}, action = left, reward = 2

LearningAgent.update(): deadline = 18, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 6), 'heading': (-1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 17, inputs = {'right': None, 'light': 'red', 'oncoming': None, 'location': (5, 6), 'heading': (-1,
0), 'left': None}, action = None, reward = 1

LearningAgent.update(): deadline = 16, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (5, 6), 'heading':
(-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 15, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (4, 6), 'heading':
(-1, 0), 'left': None}, action = forward, reward = 2

LearningAgent.update(): deadline = 14, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (3, 6), 'heading':
(-1, 0), 'left': None}, action = right, reward = 2

Environment.act(): Primary agent has reached destination!

LearningAgent.update(): deadline = 13, inputs = {'right': None, 'light': 'green', 'oncoming': None, 'location': (3, 5), 'heading':
(0, -1), 'left': None}, action = forward, reward = 12
```