## Reinforcement Learning: Putting it all Together

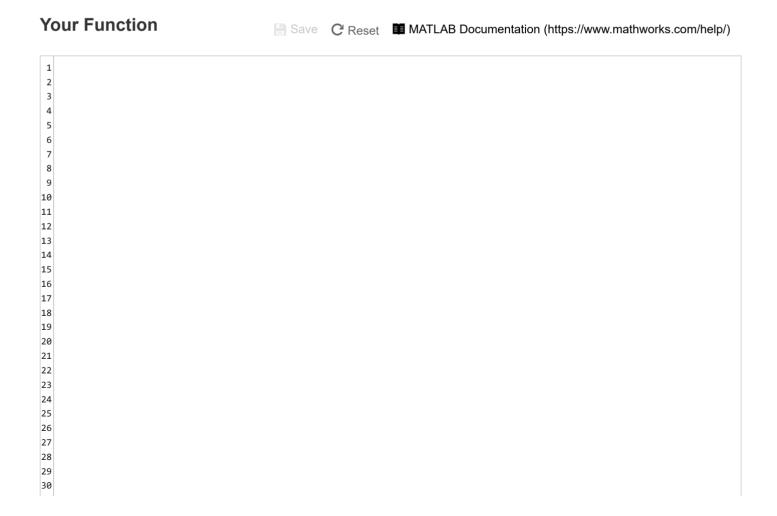
We are now ready to put the entire reinforcement learning pipeline together. Below we present a pseudo code of the learning procedure that you should implement in your function.

## Algorithm 1 Full Reinforcement Learning Pipeline 1: **Initialization:** Initialize the parameters of a network 2: **for** 1 to N **do** Set the current\_state to the start\_state 3: while current\_state is not goal\_state do 4: action=PickAction() 5: reward=GetReward() 6: new state=MakeNextState() 7: nn=DeepQLearning() 8: 9: if reward>=0 then set current\_state to new\_state 10: end if 11: end while 12: 13: end for

Note that a typical reinforcement learning procedure may take a long time. Therefore, we provide a network file that has been trained for 500 iterations but has not converged yet. You will be able to use this network file as an initialization network to speed up your training.

After running your reinforcement learning code for 100 iterations, the network should converge. Using such a network to select actions at every step, will allow the robot to reach the goal state in the minimum number of steps, which is 14 in this case.

Note that if you are working on this problem outside of the EdX environment, you should comment out or remove the line "rand('state',0);" from your PickAction.m function. If you are working inside the EdX environment, you don't need to worry about this.



```
function nn=ReinforcementLearning
32
       % A full neural network based reinforcement learning pipeline
33
34
       % Output:
35
       % - nn: a variable storing a neural network with the updated weights after full training.
36
37
       %% Initial Parameters
38
       epsilon=0.5;
39
40
       rows=7; cols=7;
41
       walls=[2 4; 3 4; 4 4; 5 4];
42
       cur_row=2; cur_col=1; rot_idx=1;
43
44
       %% Create A Start state
45
       start_S=MakeState(rows,cols,walls,cur_row,cur_col,rot_idx);
46
47
       %% Network Initialization
48
       load('RL_nn_500.mat');
49
50
       N=100;
51
52
       %% loop N times
53
       for i=1:N
54
           %% Resetting to a start state
55
```

## Code to call your function

C Reset

```
nn=ReinforcementLearning();
action_path=Prediction(nn);
number_of_actions_taken=size(action_path,1);
```

► Run Function ②

## **Previous Assessment: All Tests Passed**

Submit

Does the Network Produce a Correct Solution?