# Reinforcement Learning

## Direct MCQs for End Course Test

**Question1:** In a game with the grid size of 84 X 84 with close to 7 million possible states, if I want the agent to learn best policy by exploration which is the best algorithm to use?

1. Actor Critic
2. Policy Gradient Using REINFORCE
3. Q Learning
4. Deep Q Learning

**Answer:** Option 3. Since we want the agent to learn using exploration and it is not said that we will be using images to train the agent, Q Learning is the best algorithm to use – because it will be faster in converging due to model free approach.

**Question 2:** We want the agent to learn uncertainties in an environment, commit as less errors as possible and also maximize the reward. Which of the following options is the correct choice for such situation?

1. Dynamic Programming
2. Monte Carlo Control Methods
3. Value Iteration
4. Temporal Difference Learning

**Answer:** Option 4. Temporal Difference combines Dynamic Programming and Monte Carlo approaches because TD methods do not require a model of the environment, of its reward and next state probability distributions, there is no need to wait until the end of an episode and learn from each transition, irrespective of what subsequent actions are taken. Hence, TD approach is the correct choice.

**Question 3:** There can only be one local maxima and one global maxima when using Policy Gradients.

1. True
2. False

**Answer 3:** False. It depends upon the sampling efficiency in importance sampling. Depending upon samples generated there can be more than one local maxima. However, the global maxima remains singular.

**Question 4:** TD Learning follows the bootstrapping approach to update the action – value function whereas Monte Carlo averages out the function to stability. In what scenario is Monte Carlo better than TD Learning?

1. Monte Carlo will outperform TD Learning when off – policy methods are used as function approximators because there is a chance of non – convergence.
2. Monte Carlo is a better choice when performance assessment of multiple agents is required using policy evaluation.
3. TD Learning is always better than Monte Carlo i.e. there is no such scenario because Monte Carlo is always slower to converge than TD Learning and it has higher variance than TD Learning
4. Monte Carlo is preferred when the rules of the environment are fixed

**Answer:** Both 1 and 2. The bootstrapping process in TD Learning updates a function Q(s,a) on the next value Q(s',a') using the current estimates. In the beginning the estimates do not contain any information from any real rewards or any real transitions. This can create bias. If the learning goes well the bias will reduce asymptotically else the convergence will never happen.

**Question 5:** Which algorithm places importance on preferences of actions leading to rewards?

1. UCB
2. Gradient Bandit Algorithm
3. Epsilon – Greedy
4. Contextual Bandits

**Answer:** Option 2. In GBA respective preference of one action over another is important which results in rewards. Preference does not directly relate to rewards but it causes the agent to choose actions that produce rewards and thereby more preference is provided to that action.

**Question 6:** If there is a complex reward function and a combination of algorithms (both model based and model free) and still the learning is not converging then what could be the most likely solution?

1. Add more training data
2. It is a problem of sparse rewards and so reward shaping will resolve this issue
3. Bootstrapping via TD methods should be implemented
4. Check quality of training data, set a benchmark and stop the training if the agent is not learning anything new

**Answer:** Option 4. The agent learns what it is shown. So if everything else fails, then the only logical option is to check the quality of the training data because RL follows GIGO rule. “Garbage In – Garbage Out”.

**Question 7:** Can the entire environment be modeled before starting the exploration by an agent? If yes, which algorithm does this?

1. No, only the next state – action pair can be estimated.
2. Yes, Monte Carlo & SARSA
3. Yes, Monte Carlo & Q – Learning
4. Yes, Monte Carlo

**Answer:** Option 3. Since the environment can be stochastic, Monte Carlo will be the best suited algorithm to estimate the environment while Q – Learning can help explore those estimations and update the Q – Matrix as it is an off – policy learner.

**Question 8:** The training of an RL agent on the famous game of “Tetris” will be most accurate by which of the following options?

1. Recreation of the Tetris world in grid world and training the RL agent using any of the exploration algorithms
2. Providing gaming screens of highest scored human players and using DQN to train the agent
3. Defining the rules and using SARSA to find optimal policy
4. Using Monte Carlo and Value Iteration to train the agent

**Answer:** Option 3. The game of Tetris has limited rules of the environment that can easily be defined. Later SARSA can be used to find the optimal policy to maximize scores rather than value iteration because value iteration will promote more exploration. Here we need to exploit rather than explore. DQN will not be used because the environment is not stochastic and has very few random actions that an agent can take.

**Question 9:** Consider the set of games and approaches below:

1. Chess 1. Deep Q Networks
2. Mario 2. Epsilon Greedy
3. Pacman 3. Actor Critic
4. Crazy Taxi 4. Dynamic Programming

Which of the following option is the correct match?

* + - 1. A – 1, B – 2, C – 3, D – 4
      2. A – 4, B – 1, C – 2, D – 3
      3. A – 1, B – 3, C – 2, D – 4
      4. A – 4, B – 2, C – 1, D – 3

**Answer:** Option 2. The answer is based on how stochastic the environment is for each game. Mario has the most stochastic environment and hence Actor Critic is the right choice. In Pacman, the agent can be greedy about eating the most dots and so epsilon greedy. Chess is directly solved by dynamic programming and Crazy Taxi will be based on Deep Q Networks as the number of actions to take in an environment is limited and can be resolved by DQNs.