Quiz 5

Algorithm	Policy On/ Off	Model Based/Free
Dynamic Programming	On	Model Based
TD(λ)	On	Free
Q-Learning	Off	Free
Double Q-Learning	Off	Free
SARSA	On	Free
First Visit Montecarlo	On	Free

Algorithm	Update Function
Dynamic Programming	$V(S_t) \leftarrow E_\pi\left[R_{t+1} + \gamma V(S_{t+1})\right] = \sum_a \pi(a\|S_t) \sum_{s',r} p(s', r\|S_t, a)[r + \gamma V(s')]$
TD(λ)	$V(S_t) \leftarrow V(S_t) + \alpha\left(G_t^\lambda - V(S_t)\right)$
Q-Learning	$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_{t+1} + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)\right]$
Double Q-Learning	$Q_1(s_t, a_t) \leftarrow Q_1(s_t, a_t) + \alpha\left[r_{t+1} + \gamma Q_2(s_{t+1}, \arg\max_{a'}(Q_1(s_{t+1}, a') - Q_1(s_t, a_t)\right]$
SARSA	$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\right]$
First Visit Montecarlo	$V(s_t) \leftarrow V(s_t) + \alpha\left[G_t - V(s_t)\right]$

Algorithm	Choose Action (A)
Dynamic Programming	Greedy Policy
TD(ƛ)	policy
Q-Learning	ε - Greedy Policy
Double Q-Learning	ε - Greedy Policy
SARSA	policy
First Visit Montecarlo	ε - Soft Policy/ Exploring starts

Algorithm	Pros	Cons
Dynamic Programming	1.) Considers the probability of each action 2.) Explores all possibilities.	1.) Needs to know the entire environment 2.) Not ideal for environments with higher possibilities 3.) Not suitable for non terminating environments
TD(ƛ)	1.) Is a combination of TD(0), Monte carlo and Dynamic Programming 2.) No need to reach the terminal stage 3.) Highly suitable for terminating environments	1.) Requires a delay of n time steps before updating 2.) Needs higher Computation 3.) Needs more memory
Q-Learning	1.) Directly learns optimal policy 2.) Gives quick results	1.) Lack of proper exploration 2.) has higher per-sample variance 3.) Doesn't consider probability of actions
Double Q-Learning	1.) Double Q-learning might underestimates the action values at times, but avoids the flaw of the overestimation bias that Q-learning does 2.) In most type of problems Double Q-learning reaches good performance faster comparing to Q-learning	1.) Lack of proper exploration 2.) has higher per-sample variance 3.) Doesn't consider probability of actions 4.) High computation
SARSA	1.) A better exploration when compared to Q-Learning 2.) Lower per sample variance	1.) Takes more time to reach an optimal policy 2.)
First Visit Montecarlo	1.) Only one choice is considered at each stage 2.) Does not bootstrap from successor state's value	1.) Needs to reach the terminal stage 2.) Waits until the end of an episode

Algorithm	**Application and Comments**
Dynamic Programming	Is ideal for the known environment.
TD(ƛ)	A good combination of TD and Monte carlo. Can explore basing on the environment demand. Ideal for non terminating and unknown environments
Q-Learning	Can be applicable if the optimal path is risk free.
Double Q-Learning	Avoids the over estimation of an action. Converges faster than Q learning. Useful in environments with multiple terminal states.
SARSA	Avoids high risks and more suitable for real world applications.
First Visit Montecarlo	Is used in (quickly)terminating environments.