#Algo

i) Initialize the policy iteration, environment, discount factor (gamma) and initial policy over actions for each state.

ii) Policy Evaluation

- For a given state, calculate the value of each action.
- Update the value of current state using the calculated action values.
- Calculate the maximum change in the value function.
- Repeat if the maximum change is above the threshold.

iii) Policy Improvement:-

- For a given study calculate the value of each action.
- update the policy to choose the action with the highest value.

iv) Repeat step 2) & Step 3 until convergence.

v) Print result (optimal policy)

Exp3:-

Algo:- i) Initialize the MDP environment, discount factor (gamma) & the value function.

ii) Repeat until convergence

- For a given state, calculate the value of each action.

$$q_{\pi}(s,a) = \pi(a|s) * (R + \gamma V_{\pi}(s))$$

- Update the value function for each state using optimal action value: $V(s) = \max[Q(s,a)]\ \forall a$
- Exit if maximum change involve function is below threshold

iii) for each state, select the action that maximise the action value $\pi(s) = \text{argmax}(Q(s,a))$

iv) Print the result (optimal policy)

Exp 4:

Same as Exp2

v) print result (optimal policy, value function).

Exp5: Algorithm:-

i) Initialize the Q-learning environment, learning rate, discount factor, epsilon and the Q-function.

ii) Repeat for all the episodes till convergence.

- Decide the trade off & choose appropriate action.
- Use Bellman Equation to update the Q-function.

iii) Find the optimal policy which has the maximum Q value

iv) Print the result (Q-function & optimal policy).

Exp 6: Bellman equation :-

Algorithm:- same as MDP

iv) print (Value function & optimal policy)

Exp 7: Monte Carlo :-

Algo:-

i) Initialize Monte Carlo environment, gamma, epsilon & Q function

ii) Repeat for all the episodes:

- Decide the tradeoff and generate an episode until it terminates
- Update the Q-function using the episode history & observed returns

iii) Find the optimal policy which has maximum Q-value

iv) Print the result (Q-function + Optimal Policy)

$$Q(s,a) = Q(s,a) + \alpha\,(r + \gamma\,(\max(Q(s',a')) - Q(s,a)))$$

reward · next state · current state

learning rate · Discount factor · next action · current action