Problem

What policy would make on-policy and off-policy learning equivalent, specifically if we consider Q-learning and SARSA-learning? In other words, what policy used by an agent will make the learning based on Q-learning and SARSA-learning the same?

Problem 2

Imagine you were to design a reinforcement learning agent for playing chess. The state that the agent sees on its turn is the layout of the chess board. We can set the reward structure for the agent to be +1 for winning, -1 for losing, 0 for drawing, and 0 again for every move that does not lead to a win or loss. Such an agent will essentially learn to win. It will do so eventually after much exploration and a number of episodes, since it is always trying to maximize its expected return (cumulative rewards in the long run). What might happen if, in addition, we give a positive reward to the agent for taking its opponent's pieces as well?

Problem 3

In most real world problems with large state/action spaces, quantizing the state/action space and using tables to store/update values, e.g. the Q table, is not feasible. Can you suggest a way for reinforcement learning algorithms to generalize to arbitrarily large real valued state/action spaces? Hint: The tables are approximating a value function, i.e. mapping a state-action pair to a value. What else could be used for function approximation?