HW3 Tips and Hints Guide

Below is a consolidated "tips and hints" document that maps key excerpts from your office hours transcript to each homework question. Use these pointers to clarify your approach and check your understanding as you work through your solutions.

Problem 1: Mini-Blackjack MDP

Part (a): State Transition Diagram

- **Tip:** Although the transcript focused more on the equations, keep in mind that you need to create a diagram with seven nodes (states 0, 2, 3, 4, 5, 6, and "done").
- **Hint:** Draw edges for the "draw" action only, labeling each with the appropriate transition probabilities (each card drawn with probability 1/3).

Part (b): Optimal Actions and Values with $\gamma = 1$

- **Tip:** The discussion opened with "For part b..." and centered on how to compute state values recursively. Key points included:
  - *States 5 and 6:* Stopping is optimal because the reward equals the state's value.
  - *State Equations:* Each state's value is computed using immediate reward plus γ times the value of its successor state.
- **Hint:** You end up with a sequence of equations—one per state. You don't require a full dynamic programming approach; you can simply use previously computed values in a backward manner.

Part (c): Effect of Discount Factor on Optimal Actions

- **Tip:** The transcript discussed finding the largest γ that changes the optimal actions in states 2 and 3.
- **Hint:** For state 3, for instance, any γ below a certain threshold might reduce its value enough to alter the best action. This does not affect states 4, 5, and 6 in the same way because their values hinge on immediate stopping rewards.

Problem 2: Value Iteration (Mini-Blackjack Variation)

- **Tip:** The transcript included remarks about iterative updates, the difference between synchronous and asynchronous value iteration, and the fact that you repeatedly plug in old values to get new ones.
- **Hint:** Write the six Bellman updates for each iteration (one per non-terminal state) explicitly. Remember to use the previous iteration's values on the right-hand side.

Problem 3: Reinforcement Learning

Part (a): Monte Carlo Prediction

- **Tip:** Monte Carlo estimates come from averaging returns across complete episodes.
- **Hint:** The order in which episodes occur does not affect final state-value estimates once you average over them all.

Part (b): Temporal Difference (TD) Learning

- **Tip:** Only the state you came from is updated after each transition. The update rule is
$$V(s) \leftarrow V(s) + \alpha[r + \gamma V(s') - V(s)].$$
- **Hint:** A key comment in the transcript: "you only update the value of the state that you came from."

Part (c): Q-Learning vs SARSA Under Exploration

- **Tip:**
  - *Q-Learning* uses the max over next-state Q-values, ignoring whether exploration was used.
  - *SARSA* updates based on the actual action taken (which might be suboptimal due to exploration).
- **Hint:** Converged Q-values can differ for states where an exploratory action leads to a negative outcome, which SARSA "remembers," but Q-learning effectively ignores.

Problems 4 & 5: Bernoulli Bandits and Crawler Robot

- **Note:** These problems were not directly addressed in the office hours transcript. Refer to your class materials and the assignment description for guidance.

Final Note

This guide is meant to help you pinpoint the critical discussion points from the office hours relevant to each HW 3 problem. Combine these hints with your lecture notes and the detailed assignment instructions. Happy solving!