Name: ______________________ SUID: ______________________

Stanford University

AA228/CS238: Decision Making under Uncertainty

Autumn 2020

Prof. Mykel J. Kochenderfer • Remote • email: *mykel@stanford.edu*

MIDTERM 2 **Due date: October 23, 2020 (5pm)**

You have **90 minutes** to complete this exam. This exam is electronically timed; you do not have to keep track of your own time. To accommodate those in other time-zones and complex working situations, you may choose any 90 minute window between 5pm PDT Oct 22nd, 2020 and 5pm PDT Oct 23rd, 2020 to take the exam. Answer all questions. You may consult any material (e.g., books, calculators, computer programs, and online resources), but you may not consult other people inside or outside of the class. If you need clarification on a question, please make a private post on Piazza. **Only what is submitted prior to the deadline will be graded.**

Question 1. (1 pt) I understand the above instructions and will adhere to them.

Question 2. (4 pts) We are implementing value iteration for an MDP with 10 possible states and 5 possible actions. Assume every state-action pair has a non-zero probability of transitioning to every state - in other words, $T(s' \mid s, a) > 0$ for all s and a.

a) (2 pts) For each iteration of value iteration, where a single iteration corresponds to all states being updated, how many times will we need to evaluate the transition function?

b) (2 pts) What is one advantage of *Gauss-Seidel value iteration* over standard value iteration? Explain.

Question 3. (6 pts) We need to decide between action $A = 0$ or $A = 1$. We have a single variable C that affects our utility when we take this single action. The probability distribution over C and the utility as a function of C and the action A are given below. We would like to find the value of information for observing the value of variable C.

C	$P(C)$
0	9/10
1	1/10

A	C	$U(A, C)$
0	0	0
0	1	−20
1	0	−5
1	1	−2

a) (1 pt) Compute the optimal expected utility given C is 0.

b) (1 pt) Compute the optimal expected utility given C is 1.

c) (2 pts) What is the expected utility of taking the optimal action if we don't know the value of C?

d) (2 pts) Using your results from parts a, b, and c, compute the value of information for observing the value of the variable C.

Question 4. (8 pts) Consider an MDP with a state space with two elements and an action space with two elements. We will consider an online policy which uses branch and bound to choose the action from a given state. Assume we perform a search with a depth $d = 3$.

a) (2 pts) What is the *maximum* number of leaves that will be visited by the branch and bound algorithm? (A leaf in a tree is a node without any children.)

b) (2 pts) What is one benefit of using *branch and bound* over *forward search*? Explain.

c) (2 pts) What is one reason you would use an *online* planning method instead of *offline* planning method? Explain.

d) (BONUS: 2 pts) What is the *minimum* number of leaves that will be visited by the branch and bound algorithm? (A leaf in a tree is a node without any children.)

Question 5. (6 pts) You are a baby sea turtle trying to make it from your nest to the ocean. It's a challenging and dangerous journey of n steps, the last one being the ocean. You're new to this world so when you try taking a step there is a 50/50-chance you might slip and just stay where you are. You iteratively choose to move forward (a_1), stay put (a_2), or move backward (a_3) until you reach the ocean, where you are being rewarded with two sea turtle snacks. Each sea turtle snack corresponds to a unit of reward. You may model this as an infinite horizon problem with s_n being an absorbing state, meaning $T(s_n|s_n, a) = 1$ for all a; you cannot leave the ocean once you've entered it.

Your current policy is to stay put, unless you are one step away from the ocean s_{n-1}, where you would move forward to reach the ocean (s_n). Let's perform one step of policy optimization:

a) (4 pts) With $\gamma = 0.5$ and $n = 10$, evaluate your current policy.
Hint: If you choose to invert a matrix. You can use the following link https://matrix.reshish.com/inverse.php *or any other tool of your choice.*

b) (2 pts) Compute your policy improvement: state your resulting action value function $Q(s, a)$ and new policy $\pi(s)$.

For ease of grading *please* put your $Q(s, a)$ in an $n \times 3$ matrix. The first row should correspond to state 1, the second to state 2, and so on. The first column should correspond to a_1, the second to a_2, and the third to a_3.

Question 6. (3 pts) What is one advantage of the cross entropy method over Hooke-Jeeves for policy optimization? Explain.

Question 7. (BONUS: 2 pts) Suppose we are using policy gradient optimization. If all trajectories used have a reward of 0, what effect will the gradient ascent step have on the policy? Explain.