CSE 473 Midterm Exam – Spring 2019

Name:

This exam is take home and is due on **Mon May 13**. You can submit it online (see the message board for instructions) before midnight or hand it in at the beginning of class (and we will scan it for you). This exam should not take significantly longer than 3 hours to complete if you have already carefully studied all of course material. Studying while taking the exam may take longer. :)

This exam is open book and open notes, but you must complete all of the work yourself with no help from others. Please feel free to post clarification questions to the class message board, but please do not discuss solutions there.

Partial Credit: If you show your work and *briefly* describe your approach to the longer questions, we will happily give partially credit, where possible. We reserve the right to take off points for overly long answers. Please do not just write everything you can think of for each problem.

Name: Please do not forget to write your name in the space above!

Question 1 – True/False – 30 points

Circle the correct answer each True / False question.

1. True / False – Reflex agents (agents that act according to a precomputed look up table) cannot act optimally (in terms of maximizing total expected reward over time). (3 pt)

2. True / False – Minimax is optimal against any opponent. (3 pt)

3. True / False – Greedy search can take longer to terminate than uniform cost search. (3 pt)

4. True / False – Uniform cost search with costs of 1 for all transitions is the same as depth first search. (3 pt)

5. True / False – Alpha-Beta pruning can introduce errors during mini-max search. (3 pt)

6. True / False – Each state can only appear once in a search tree. (3 pt)

7. True / False – Policy iteration always find the optimal policy, when run to convergence. (3 pt)

8. True / False – Higher values for the discount (γ) will, in general, cause value iteration to converge more slowly. (3pt)

9. True / False – For MDPs, adapting the policy to depend on the previous state, in addition to the current state, can lead to higher expected reward. (3pt)

10. True / False – Graph search can sometimes expand more nodes than tree search. (3pt)

Question 2 – Short Answer – 30 points

These short answer questions can be answered with a few sentences each.

1. Short Answer – Briefly describe the relationship between admissible and consistent heuristics. When would you use each, and why? (5 pts)

2. Short Answer – Briefly describe when you would use Alpha-beta pruning in minimax search. (5 pts)

3. Short Answer – Is it always possible to solve a CSP faster with cutset conditioning, as compared to forward checking with arc consistency. If so, briefly justify your answer. If not, describe the conditions under which cutset conditioning would be preferred. (5 pts)

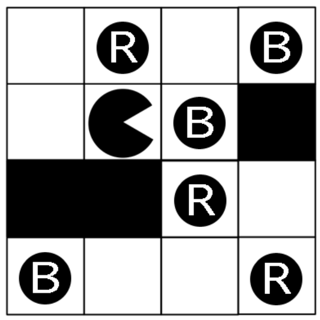
4. Short Answer – Briefly describe the difference between UCS and A* search. When would you prefer to use each, and why? (5 pts)

5. Short Answer – Write down the time complexity (big O notation) for checking K-consistency in CSPs, as a function of K and any other relevant properties of the CSP. Be sure to define all of the constants you use. (5 pts)

6. Short Answer – Briefly describe the difference between value iteration and policy iteration. Describe conditions under which one algorithm might be preferred to the other, in practice. (5 pts)

Question 3 – Ordered Pacman Search – 25 points

Consider a new Pacman game where there are two kinds of food pellets, each with a different color (red and blue). Pacman has peculiar eating habits; he strongly prefers to eat pellets in an order where the colors alternate (e.g. red, blue, red, etc.). If Pacman eats two pellets of the same color consecutively, he will incur a cost of 100. Otherwise, as before, there is a cost of 1 for each step and the goal is to eat all the dots. There are K red pellets and K blue pellets, and the dimensions of the board are N by M.



$K = 3$, $N = 4$, $M = 4$

1. Give a non-trivial upper bound on the size of the state space required to model this problem. Briefly describe your reasoning. [10 pts]

2. Give a non-trivial upper bound on the branching factor of the state space. Briefly describe your reasoning. [5 pts]

3. Name a search algorithm pacman could execute to get the optimal path? Briefly justify your choice (describe in one or two sentences) [5 pts]

4. Give a non-trivial admissible heuristic for this problem. Briefly describe why it is admissible. [5 pts]

Question 4 - Pacman CSP – 20 points

After years of struggling through mazes, Pacman has finally made peace with the ghosts, Blinky, Pinky, Inky, and Clyde, and invited them to live with him and Ms. Pacman. The move has forced Pacman to change the rooming assignments in his house, which has 6 rooms. He has decided to figure out the new assignments with a CSP in which the variables are Pacman **(P)**, Ms. Pacman **(M)**, Blinky **(B)**, Pinky **(K)**, Inky **(I)**, and Clyde **(C)**, the values are which room they will stay in, from 1-6, and the constraints are:

i) No two agents can stay in the same room
ii) **P** $>$ 3
iii) **K** is less than **P**
iv) **M** is either 5 or 6
v) **P** $>$ **M**
vi) **B** is even
vii) **I** is not 1 or 6
viii) $|\mathbf{I}\text{-}\mathbf{C}| = 1$
ix) $|\mathbf{P}\text{-}\mathbf{B}| = 2$

1. Unary constraints restrict the values of individual variables in the CSP. On the grid below cross out the values that are eliminated by enforcing unary constraints. [5 pts]

P	1	2	3	4	5	6
B	1	2	3	4	5	6
C	1	2	3	4	5	6
K	1	2	3	4	5	6
I	1	2	3	4	5	6
M	1	2	3	4	5	6

2. According to the Minimum Remaining Value (MRV) heuristic, which variable would be assigned a value first after enforcing unary constraints? [5 pts]

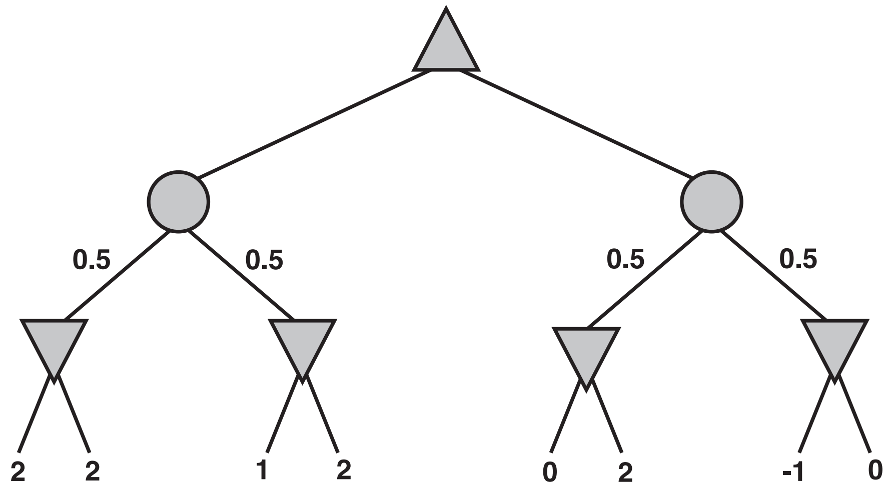
3. For the purposes of decoupling this problem from your solution to the previous problem, assume we choose to assign P first, and assign it the value 6. What are the resulting domains after enforcing unary constraints (from part i) and running forward checking for this assignment? [5 pts]

P						6
B	1	2	3	4	5	6
C	1	2	3	4	5	6
K	1	2	3	4	5	6
I	1	2	3	4	5	6
M	1	2	3	4	5	6

4. Draw the constraint graph for this problem. Based on the overall graph structure, which algorithm would you use to solve the CSP, and why? [5pts]

Question 5 – Game Trees – 30 points

Consider the following game tree, which has min (down triangle), max (up triangle), and expectation (circle) nodes:



1. In the figure above, label each tree node with its value (a real number). [7 pts]

2. In the figure above, circle the edge associated with the optimal action at each choice point. [7 pts]

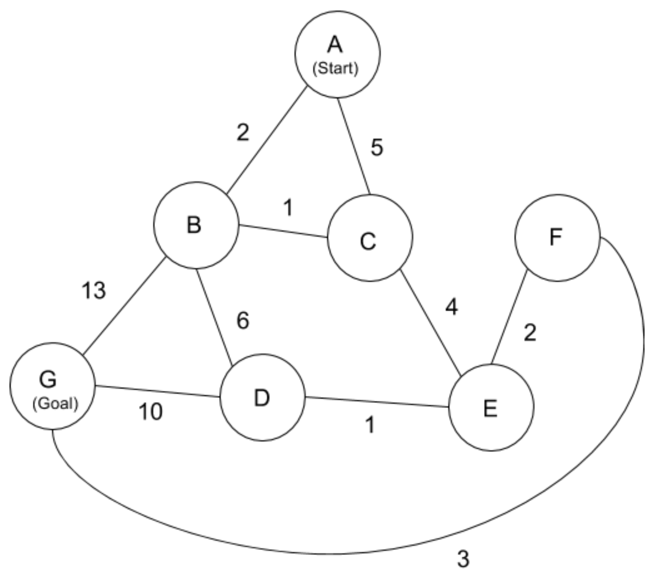
3. If we knew the values of the first six leaves (from left), would we need to evaluate the seventh and eighth leaves? Why or why not? [5 pts]

4. Suppose the values of leaf nodes are known to be in the range $[-2, 2]$, inclusive. Assume that we evaluate the nodes from left to right in a depth first manner. Can we now avoid expanding the whole tree? If so, why? Circle all of the nodes that would need to be evaluated (include them all if necessary). [11 pts]

Question 6 – Graph Search – 30 points

Given the state graph below, run each of the following algorithms and list the order that the nodes are expanded (a node is considered expanded when it is dequeued from the fringe). The values next to each edge denote the cost of traveling between states. All edges are bidriectional, but you should be sure to use graph search and not tree search.

When expanding the children of a node, you should add them to the fringe in alphabetical order. Every solution should always start with the start node and end with the goal node.



1. Breadth first search [5 pts]

2. Depth first search [5 pts]

3. Iterative deepening [5 pts]

4. Uniform cost search [5 pts]

Now, consider the following two heuristics:

State s	H1(s)	H2(s)
A (start)	10	12
B	8	11
C	7	8
D	4	4
E	3	4
F	2	3
G (goal)	0	0

5. Provide the expansion ordering for A* search with heuristic H2. [5 pts]

6. List which, if any, of the two heuristics are admissible [2.5 pts]

7. List which, if any, of the two heuristics are consistent [2.5 pts]

Question 7 – Reset MDP and Bellman Equations – 25 points

Let M be an arbitrary MDP with start state s_0. Define M' to be a modification of M, where every time the agent takes an action it has some chance (with probability α) of randomly teleporting back to s_0. The other $1 - \alpha$ percent of the time it transitions according to the function $T(s, a, s')$ defined in M.

1. Write down a set of Bellman equations for the reset MDP M' by defining $V^*(s)$, $Q^*(s, a)$ and $\pi^*(s)$. Be sure to include the reset probability α and the discount γ. You can also use all of the other functions and sets defined in M, as needed. [25 pts]

2. Define V^* to be the optimal value function for M and $V^{*\prime}$ to be the optimal value function for M'. Will V^* and $V^{*\prime}$ be the same function? If so, briefly justify your answer. If not, briefly describe how they differ. [5 pts]

3. Define π^* to be the optimal policy for M and $\pi^{*\prime}$ to be the optimal policy for M'. Will π^* and $\pi^{*\prime}$ be the same function? If so, briefly justify your answer. If not, briefly describe how they differ. [5 pts]