CS 188
Fall 2009

Introduction to
Artificial Intelligence

Midterm Exam

INSTRUCTIONS

- You have 3 hours.
- The exam is closed book, closed notes except a one-page crib sheet.
- Please use non-programmable calculators only.
- Mark your answers ON THE EXAM ITSELF. If you are not sure of your answer you may wish to provide a *brief* explanation. All short answer sections can be successfully answered in a few sentences at most.

Last Name	
First Name	
SID	
Login	
GSI	
Section Time	
All the work on this exam is my own. **(please sign)**	

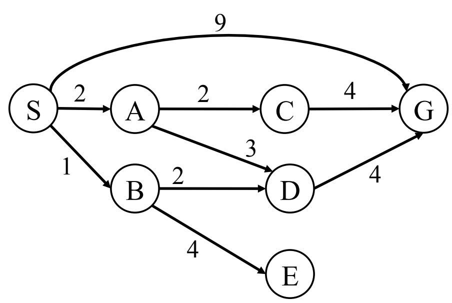
For staff use only

Q. 1	Q. 2	Q. 3	Q. 4	Q. 5	Q. 6	Total
/11	/10	/12	/12	/13	/12	/70

THIS PAGE INTENTIONALLY LEFT BLANK

1. (11 points) Search

Consider the following search problem, represented as a graph. The start state is S and the only goal state is G. Note that the following problems variously reference both tree search and graph search. For questions which require a heuristic, use the one given below.



Heuristic

S	A	B	C	D	E	G
6	0	6	4	1	10	0

(a) (1 pt) What path will BFS *tree search* return?

(b) (1 pt) What path will UCS *tree search* return?

(c) (1 pt) What path will UCS *graph search* return?

(d) (2 pt) For UCS *graph search*, write the ordered sequence of states put on the closed list up to termination.

(e) (1 pt) What path will greedy *tree search* return?

(f) (1 pt) What path will A* *tree search* return?

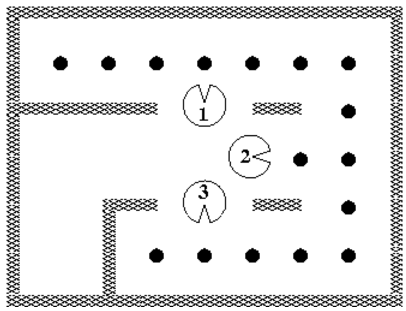
(g) (1 pt) Name a path that UCS expands (pops off the queue) but A* does not.

(h) (2 pt) What path will A* *graph search* return? *Hint: be careful.*

(i) (1 pt) Explain why your answer to part (h) is reasonable.

2. (10 points) Cooperative Pac-Family

Pacman is trying eat all the dots, but he now has the help of his family! There are initially k dots, at positions $(f_1, \ldots f_k)$. There are also n Pac-People, at positions $(p_1, \ldots, p_n)$; initially, all the Pac-People start in the bottom left corner of the maze. Consider a search problem in which all Pac-People move *simultaneously*; that is, in each step each Pac-Person moves into some adjacent position (N, S, E, or W, no STOP). Note that any number of Pac-People may occupy the same position.



(a) (3 pt) Define the state space of the search problem.

(b) (1 pt) Give a reasonable upper bound on the size of the state space for a general r by c grid.

(c) (1 pt) What is the goal test?

(d) (1 pt) What is the maximum branching factor of the successor function in a general grid?

(e) (4 pt) Circle the admissible heuristics below (-1/2 point for each mistake.)

i. $h_1(s) = 0$

ii. $h_2(s) = 1$

iii. $h_3(s) =$ number of remaining food / n

iv. $h_4(s) = \max_i \max_j \text{manhattan}(p_i, \text{food}_j)$

v. $h_5(s) = \max_i \min_j \text{manhattan}(p_i, \text{food}_j)$

vi. $h_6(s) = \min_i \max_j \text{manhattan}(p_i, \text{food}_j)$

vii. $h_7(s) = \min_i \min_j \text{manhattan}(p_i, \text{food}_j)$

viii. $h_8(s) = \max(h_3, h_7)$

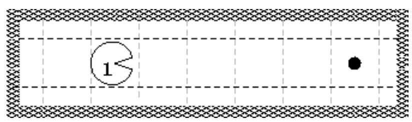
ix. $h_9(s) = \min(h_3, h_4)$

x. $h_{10}(s)$ = Solve the single-agent food search problem for each Pac-Person individually from their current position. Return the minimum of these divided by n.

3. (12 points) Competitive Pac-Family

Cooperation only lasts so long. Suddenly, the Pac-People are competing to see who can eat the most food. Specifically, each Pac-Person's score is the number of dots they have eaten minus the number of dots that any other Pac-Person has eaten. Whoever has the highest score when all the food is gone is the winner. Now, they *take turns*, one Pac-Person moving each turn, p_1 first, then p_2, etc. Unlike in project 2, there is no -1 point "cost of living" per round.

First consider the simple 1 player case, in the board below. Pacman can move either *West* (left) or *East* (right) and is using limited-depth minimax search with a basic score evaluation function to choose his next move. He is 5 *East* moves away from the food.



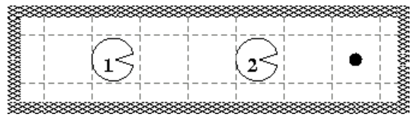
(a) (2 pt) For what search depths will *East* be an optimal action (i.e. have the highest minimax value)?

(b) (2 pt) For what search depths will *West* be an optimal action (i.e. have the highest minimax value)?

(c) (1 pt) Now, Pacman is using an evaluation function of $\text{score} + 0.5/\text{dist} - \text{to} - \text{closest} - \text{dot}$. For what search depths is *East* optimal?

(d) (1 pt) Given the new evaluation function in (c), for what search depths is *West* optimal?

A second Pac-Person enters the game. Consider the board below, where p_1 is 5 moves from the food and p_2 is 2 away. It is p_1's move. Here, the evaluation function p_1 uses is again simply its score. Remember, p_1's score is the number of dots p_1 has eaten minus the number of dots p_2 has eaten (and so p_1 and p_2 always have opposite scores).



(e) **(1 pt)** If p_1 uses minimax search to depth 10, what will be the minimax value of root node in his search tree?

(f) **(1 pt)** Now suppose p_1 knows that p_2 (a Pac-Baby) just moves randomly. If p_1 uses a depth 10 expectimax search, what will be his optimal action or actions?

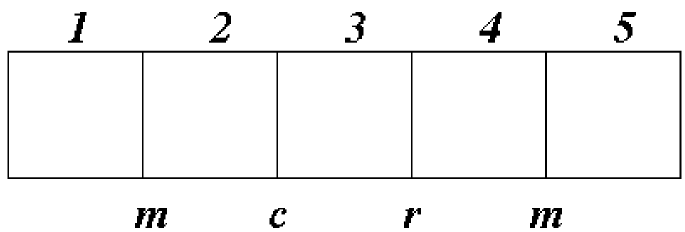
(g) **(2 pt)** A third Pac-Person wants to play, too, so they move to a larger board. Assuming all three players are playing optimally, why is it a mistake for p_1 to use minimax search where p_2 and p_3 are modeled as min nodes?

(h) **(2 pt)** Five players! Suppose p_1 and p_2 are playing against p_3 and p_4 (teams of 2, each player wanting to maximize their team's score). p_5 (a randomly moving Pac-Baby) is also playing. They move in numerical order. Sketch a single ply of the search tree from p_1's perspective (you don't need to draw all possible children, just indicate who each layer represents and what the layer types are).

4. (12 points) CSP: Criminal Search and Prosecution

You are a detective in charge of bringing down drug dealers (D). A tip has led you to a small apartment complex where you believe one or more D might be hiding out.

There are five apartments in a row. Each apartment could contain a drug dealer D or could contain innocent people: adults (A), families with babies (B), or with teenagers (T). Before you break down a door, you need to be absolutely sure that a dealer D is inside, otherwise you could get sued for police suboptimality.



To help you narrow down where drug dealers D might be (if any are there at all!), you use the fact that different people make different noises. Every time you walk between two apartments, you can hear the louder of the two noises that are being made in those apartments. The loudest people are teenagers T, who blast music (m), the next loudest are babies B who cry (c), the next loudest is the drug dealer D, who makes a rustling sound (r), and the quietest people are adults A, who are totally silent (s). For example, if there were a baby in one house and a teenager in next, you would hear music (m) when standing between those apartments.

Walking by the five apartments, you hear the noises shown in the diagram above. You decide to try solving this problem as a CSP.

(a) (1 pt) What are the variables and domains in this CSP? (Do not use a formulation where the noises are variables; they'll show up as constraints.)

(b) (4 pt) Write down the binary and/or unary constraints implied by the noises shown above at the apartment boundaries.

(c) (2 pt) You decide to narrow down your domains by enforcing arc-consistency. What are the remaining domains of each variable after arc consistency is enforced.

(d) (1 pt) If you wanted to find a solution at this point, what variables could the MRV (minimum remaining values) ordering heuristic tell you to assign first?

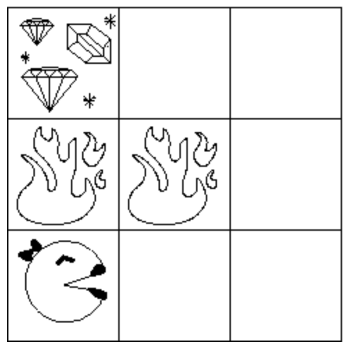
(e) (1 pt) List all solutions to this CSP or state that none exist.

(f) (1 pt) Is it ok for you to break into apartment 4? Justify your answer.

(g) (2 pt) Imagine that you cannot actually solve CSPs yourself in the field. However you *can* formulate them and give them to the police department's *CSP detector*. A CSP detector takes in your variables, domains, and unary and binary constraints, and outputs *true* if at least one solution exists. Using a detector like this, describe how you could prove that some apartment k was safe to storm into?

5. (13 points) MDPs: Treasure Hunting

While Pacman is out collecting all the dots from `mediumClassic`, Ms. Pacman takes some time to go treasure hunting in the Gridworld island. Ever prepared, she has a map that shows where all the hazards are, and where the treasure is. From any unmarked square, Ms. Pacman can take the standard actions (N, S, E, W), but she is surefooted enough that her actions always succeed (i.e. there is no movement noise). If she lands in a hazard (H) square or a treasure (T) square, her only action is to call for an airlift (X), which takes her to the terminal 'Done' state, receiving a reward of -64 if she's escaping a hazard, but +128 if she's running off with the treasure. There is no "living reward."



(a) (2 pt) What are the optimal values, V^* of each state in the above grid if $\gamma = 0.5$?

(b) (1 pt) What's the optimal policy?

Call this policy π_0. Ms. Pacman realizes that her map might be out of date, so she decides to do some Q-learning to see what the island is really like. Because she thinks π_0 is close to correct, she decides to Q-learn while following an ϵ-random policy based on (b). Specifically, with probability ϵ she chooses amongst the available actions uniformly at random. Otherwise, she does what π_0 recommends. Call this policy π_ϵ.

An ϵ-random policy like π_ϵ is an example of a *stochastic* policy, which assigns probabilities to actions rather than recommending a single one. A stochastic policy can be written as $\pi(s, a)$, the probability of taking action a when the agent is in state s.

(c) **(2 pt)** Write out a modified Bellman equation for policy evaluation when the policy $\pi(s, a)$ is stochastic.

(d) **(1 pt)** If Ms. Pacman's map is correct what relationship will hold for all states?

i. $V^{\pi_0} \geq V^{\pi_\epsilon}$

ii. $V^{\pi_0} = V^{\pi_\epsilon}$

iii. $V^{\pi_0} \leq V^{\pi_\epsilon}$

It turns out that Ms. Pacman's map is mostly correct, but some of the fire pits may have fizzled out and become regular squares! Thus, when she starts Q-learning, she observes the following episodes:
[(0, 0), N, 0, (0, 1), N, 0, (0, 2), X, 128, Done]
[(0, 0), N, 0, (0, 1), N, 0, (0, 2), X, 128, Done]
[(0, 0), N, 0, (0, 1), E, 0, (1, 1), X, -64, Done]

(e) **(2 pt)** What are Ms. Pacman's Q-values after observing these episodes? Assume that she initialized her Q-values all to 0 (you only have to write the Q-values that aren't 0) and used a learning rate of 1.0.

(f) **(2 pt)** In most cases, a learning rate of 1.0 will result in a failure to converge. Why is it safe for Ms. Pacman to use a learning rate of 1.0?

(g) **(1 pt)** Based on your knowledge about the structure of the maze and the episodes Ms. Pacman observed, what are the *true* optimal values of each state?

6. (12 points) Probability: Mini Power Pellets

Note: For this problem, any answers that require division can be left written as a fraction.

PacLabs has just created a new type of mini power pellet that is small enough for Pacman to carry around with him when he's running around mazes. Unfortunately, these mini-pellets don't guarantee that Pacman will win all his fights with ghosts, and they look just like the regular dots Pacman carried around to snack on.

Pacman just ate a snack (P), which was either a mini-pellet ($+p$), or a regular dot ($-p$), and is about to get into a fight (W), which he can win ($+w$) or lose ($-w$). Both these variables are unknown, but fortunately, Pacman is a master of probability. He knows that his bag of snacks has 5 mini-pellets and 15 regular dots. He also knows that if he ate a mini-pellet, he has a 70% chance of winning, but if he ate a regular dot, he only has a 20% chance.

(a) (2 pt) What is $P(+w)$, the marginal probability that Pacman will win?

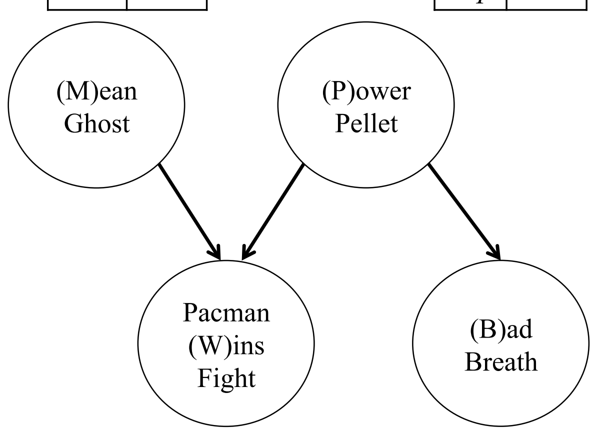
(b) (1 pt) Pacman won! Hooray! What is the conditional probability $P(+p \mid +w)$ that the food he ate was a mini-pellet, given that he won?

Pacman can make better probability estimates if he takes more information into account. First, Pacman's breath, B, can be bad ($+b$) or fresh ($-b$). Second, there are two types of ghost (M): mean ($+m$) and nice ($-m$). Pacman has encoded his knowledge about the situation in the following Bayes' Net:

$P(M)$	
$+m$	0.50
$-m$	0.50

$P(P)$	
$+p$	0.25
$-p$	0.75

$P(W \mid M, P)$			
$+m$	$+p$	$+w$	0.60
$+m$	$+p$	$-w$	0.40
$+m$	$-p$	$+w$	0.10
$+m$	$-p$	$-w$	0.90
$-m$	$+p$	$+w$	0.80
$-m$	$+p$	$-w$	0.20
$-m$	$-p$	$+w$	0.30
$-m$	$-p$	$-w$	0.70



$P(B \mid P)$		
$+p$	$+b$	0.80
$+p$	$-b$	0.20
$-p$	$+b$	0.40
$-p$	$-b$	0.60

(c) (2 pt) What is the probability of the atomic event $(-m, +p, +w, -b)$, where Pacman eats a mini-pellet and has fresh breath before winning a fight against a nice ghost?

(d) (3 pt) Which of the following conditional independence statements are guaranteed to be true by the Bayes' Net graph structure?

i. $W \perp\!\!\!\perp B$
ii. $W \perp\!\!\!\perp B|P$
iii. $M \perp\!\!\!\perp P$
iv. $M \perp\!\!\!\perp P|W$
v. $M \perp\!\!\!\perp B$
vi. $M \perp\!\!\!\perp B|P$
vii. $M \perp\!\!\!\perp B|W$

For the remainder of this question, use the half of the joint probability table that has been computed for you below:

$P(M, P, W, B)$				
$+m$	$+p$	$+w$	$+b$	0.0800
$+m$	$+p$	$+w$	$-b$	0.0150
$+m$	$+p$	$-w$	$+b$	0.0400
$+m$	$+p$	$-w$	$-b$	0.0100
$+m$	$-p$	$+w$	$+b$	0.0150
$+m$	$-p$	$+w$	$-b$	0.0225
$+m$	$-p$	$-w$	$+b$	0.1350
$+m$	$-p$	$-w$	$-b$	0.2025

(e) (1 pt) What is the marginal probability, $P(+m, +b)$ that Pacman encounters a mean ghost and has bad breath?

(f) (1 pt) Pacman observes that he has bad breath and that the ghost he's facing is mean. What is the conditional probability, $P(+w \mid +m, +b)$, that he will win the fight, given his observations?

(g) (2 pt) Pacman's utility is +10 for winning a fight, -5 for losing a fight, and -1 for running away from a fight. Pacman wants to maximize his expected utility. Given that he has bad breath and is facing a mean ghost, should he stay and fight, or run away? Justify your answer numerically!