CS 188 Fall 2020 — Introduction to Artificial Intelligence — Final

- You have approximately 170 minutes.
- The exam is open book, open calculator, and open notes.
- For multiple choice questions,
  - ☐ means mark **all options** that apply
  - ◯ means mark a **single choice**

First name	
Last name	
SID	

For staff use only:

Q1.	Shrektacular Swamp	/16
Q2.	Pac-Pact	/16
Q3.	Tom and Jerry, Continued	/11
Q4.	Mesut-Bot Going to Class	/19
Q5.	Reinforcement Learning on Belief States	/11
Q6.	Pac-mate or Pac-poster	/10
Q7.	Value of Pacman Information	/17
	Total	/100

THIS PAGE IS INTENTIONALLY LEFT BLANK

SID: _____

Q1. [16 pts] Shrektacular Swamp

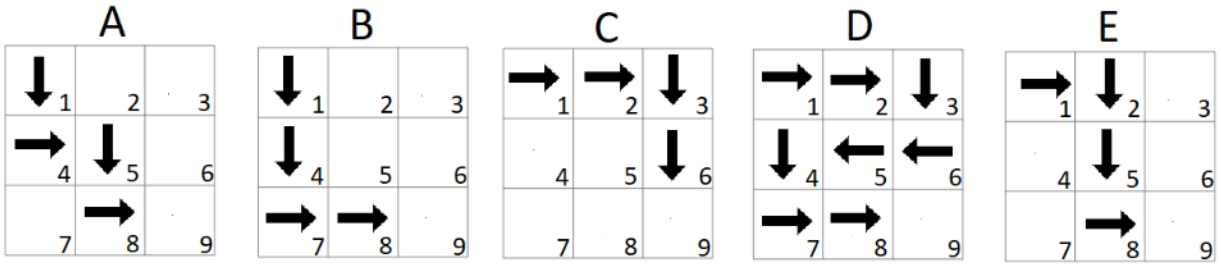
Shrek finds himself in the 3x3 grid world below. He starts in square 1 and he wants to reach his swamp in square 9. His 4 available actions are to move up, left, right, and down, from one numbered square to another. Actions that move Shrek out of the grid are not allowed.

1	2	3
4	5	6
7	8	9

(a) Shrek decides to plan his route using search algorithms. His state space contains 9 total states, one for each square in the grid.

There are 5 candidate paths from state 1 to state 9 below, labeled (A through E). For each search algorithm below select all paths that the algorithm could possibly return under some tie breaking system.

Note for A* algorithms, n is the integer label of the square, the cost we are trying to minimize is the total number of moves, and we only update a state's value in the fringe if the value in the fringe is strictly greater than the value we are updating to.



(i) [1 pt] DFS Graph Search
☐ A ☐ B ☐ C ☐ D ☐ E ○ None of the paths

(ii) [1 pt] BFS
☐ A ☐ B ☐ C ☐ D ☐ E ○ None of the paths

(iii) [2 pts] A* with heuristic $h(n) = 10 * n$
☐ A ☐ B ☐ C ☐ D ☐ E ○ None of the paths

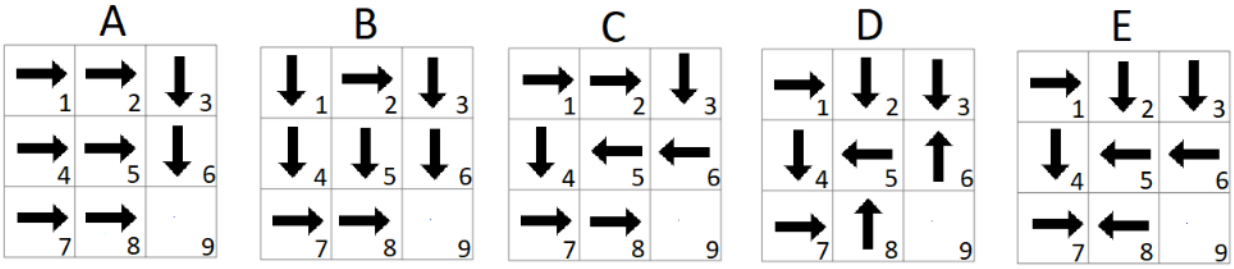
(iv) [2 pts] A* with heuristic $h(n) = 10 * (10 - n)$
☐ A ☐ B ☐ C ☐ D ☐ E ○ None of the paths

(v) [2 pts] A* with heuristic $h(n) = n \mod 3$ (where $x \mod y$ is the remainder when x is divided by y).
☐ A ☐ B ☐ C ☐ D ☐ E ○ None of the paths

(b) Lord Farquaad wants to take over Shrek's swamp so he decides to navigate the same grid world. However, he decides to frame the problem as an MDP with 9 states and 4 actions. The transitions are deterministic and work the same as the previous part. Farquaad is forbidden from taking any action that attempts to move him out of the grid. State 9 is a terminal state so no more actions can be taken from there.

There are 5 candidate policies below. For each reward function and discount factor pair below select all candidate policies that will get you the optimal reward for all 8 non-terminal states.

Note, $I(condition)$ is an indicator function that is equal to 1 if the $condition$ is true and it is equal to 0 otherwise.



(i) [1 pt] $R(s, a, s') = I(a = right) + 10 * I(s' = 9), \quad \gamma = .5$

☐ A ☐ B ☐ C ☐ D ☐ E ○ None of the policies

(ii) [1 pt] $R(s, a, s') = -1, \quad \gamma = .75$

☐ A ☐ B ☐ C ☐ D ☐ E ○ None of the policies

(iii) [2 pts] $R(s, a, s') = I(a = right) + 10 * I(s' = 9) - I(a = left), \quad \gamma = 1$

☐ A ☐ B ☐ C ☐ D ☐ E ○ None of the policies

(iv) [2 pts] $R(s, a, s') = I(a = right) + I(a = left), \quad \gamma = .75$

☐ A ☐ B ☐ C ☐ D ☐ E ○ None of the policies

(v) [2 pts] $R(s, a, s') = 10 * I(s' \neq 9) + 15 * I(s' = 9), \quad \gamma = .5$

☐ A ☐ B ☐ C ☐ D ☐ E ○ None of the policies

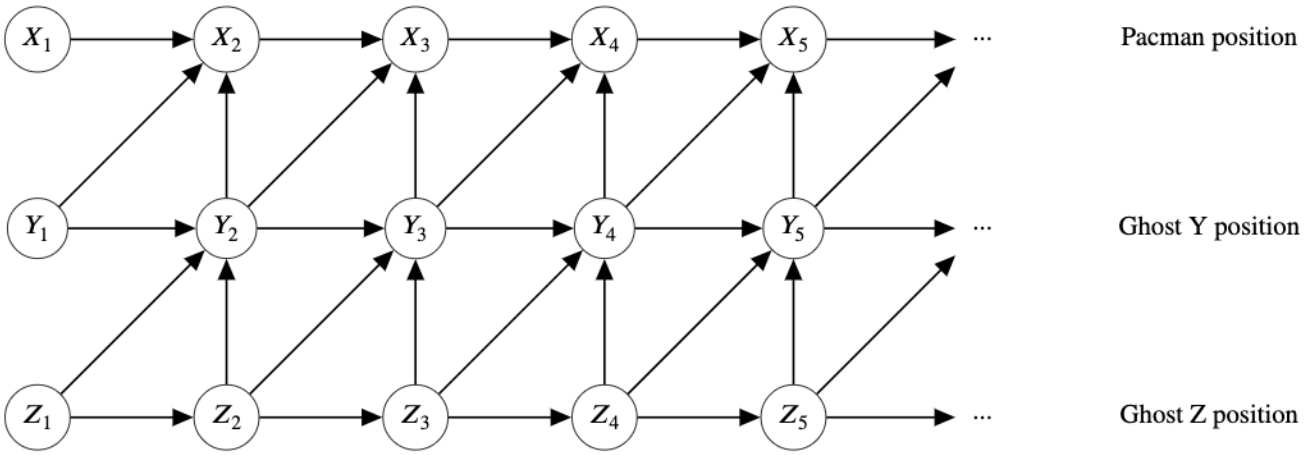
SID: _____

Q2. [16 pts] Pac-Pact

Pacman and his two ghost buddies are driving down a 3 lane highway, each in their own lane. They want to stay next to each other for the entirety of the n timesteps of their roadtrip. For any timestep t, let X_t denote the position of Pacman, Y_t the position of Ghost Y, and Z_t the position of Ghost Z.

- Pacman and his ghost buddies start on the three lanes in random positions.
- At each timestep, each agent either accelerates or decelerates by a controllable amount.
- We model this problem such that during each timestep t, Ghost Z acts first, Ghost Y acts second, and Pacman acts third.
- Each agent's position at time t is influenced by their position in the previous timestep, $t-1$.
- Ghost Y's position at time t (Y_t) is influenced by Ghost Z's position at time $t-1$ (Z_{t-1}) and t (Z_t).
- Pacman's position at time t is influenced by Ghost Y's position at time $t-1$ (Y_{t-1}) and t (Y_t).

This gives us the Bayes Net below.



(a) First we analyze the independence assumptions of our Bayes Net.

(i) [2 pts] Which of the following independence relations are guaranteed?

- ☐ $Z_4 \perp\!\!\!\perp Y_3$
- ☐ $X_t \perp\!\!\!\perp X_{t-3} | X_{t-1}$ for $4 \leq t \leq n$
- ☐ $X_t \perp\!\!\!\perp X_{t-3} | X_{t-1}, Y_{t-1}$ for $4 \leq t \leq n$
- ☐ $X_t \perp\!\!\!\perp X_{t-3} | X_{t-1}, Y_{t-1}, Z_{t-1}$ for $4 \leq t \leq n-1$
- ○ None of the above

(ii) [2 pts] If we are trying to determine Pacman's position on the highway from timesteps 1 to 4, and we are given Ghost Y's position from timesteps 1 to 4, which of the following are guaranteed to not give us additional information to solve our problem?

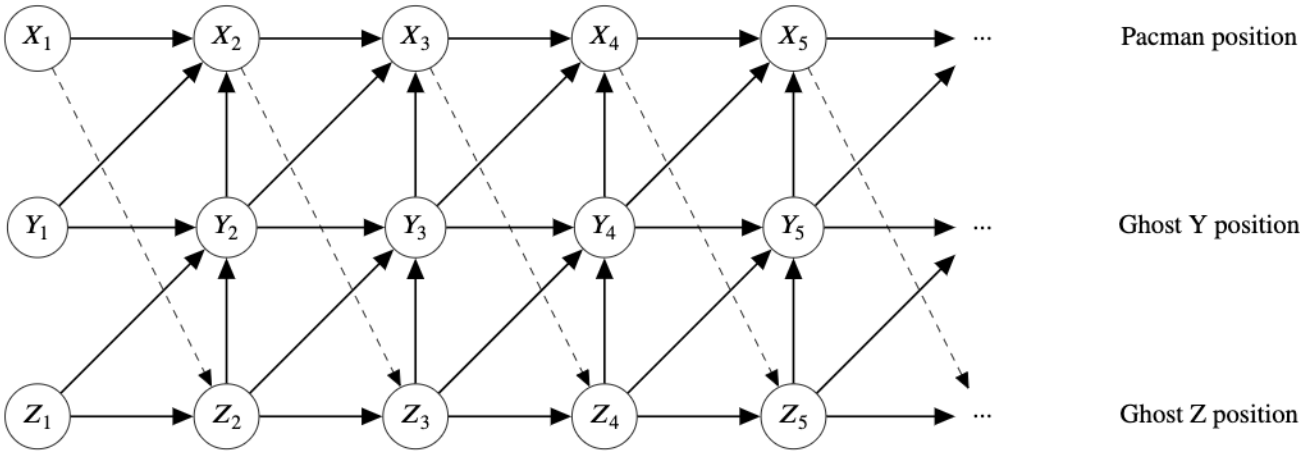
- ☐ Ghost Z's position at timestep 3
- ☐ Ghost Z's position at timestep 5
- ☐ Ghost Y's position at timestep 5
- ☐ Pacman's position at timestep 5
- ○ None of the above

(iii) [1 pt] What is the minimal number of variables that must be given for Pacman's position at time t to be independent of Ghost Z's position at time t, for $t \geq 2$? _____

You realize that the Bayes Net above does not accurately reflect the interactions between Pacman and his ghost buddies. You notice that:

- Ghost Z takes cues from Pacman with probability p, so Ghost Z's position at any time t depends on Pacman's position at time $t-1$ with probability $p < 1$.

Taking this into account, you modify the Bayes Net as shown below, with dotted edges denoting those that exist with probability $p < 1$.



(b) **(i)** [1 pt] If the entire roadtrip was n timesteps long, what is the expected number of timesteps where Ghost Z takes into account Pacman's position before moving?

- ◯ $n-1$
- ◯ $(n-1)p$
- ◯ $\frac{n-1}{p}$
- ◯ $\frac{n-1}{1-p}$
- ◯ $(n-1)(1-p)$
- ◯ None of the above

(ii) [1 pt] What is the probability we are guaranteed that $X_1 \perp\!\!\!\perp X_3 | X_2$ based on conditional independence assumptions encoded in the Bayes Net?

- ◯ 0
- ◯ p
- ◯ p^2
- ◯ $p(1-p)$
- ◯ $2p(1-p)$
- ◯ $(1-p)^2$
- ◯ $1-p$
- ◯ 1
- ◯ None of the above

(iii) [1 pt] What is the probability we are guaranteed that $X_1 \perp\!\!\!\perp X_t | X_2, Y_1, Y_2$, for any $3 < t < n$ based on conditional independence assumptions encoded in the Bayes Net?

- ◯ 0
- ◯ p
- ◯ p^2
- ◯ $p(1-p)$
- ◯ $p(1-p)^{t-1}$
- ◯ $p^{t-1}(1-p)$
- ◯ $2p(1-p)$

- ◯ $(1-p)^2$
- ◯ $1-p$
- ◯ 1
- ◯ None of the above

(iv) [2 pts] What is the probability we are guaranteed that $X_1 \perp\!\!\!\perp X_t | X_k, Y_{k-1}, Y_k$, for any k, t where $2 < k < t < n$ based on conditional independence assumptions encoded in the Bayes Net?

- ◯ 0
- ◯ $p^{k-1}(1-p)^{t-k}$
- ◯ $p^{t-k}(1-p)^{k-1}$
- ◯ $p^{k}(1-p)^{t-k}$
- ◯ $p^{t-k}(1-p)^{k}$
- ◯ $\binom{t-1}{k} p^{k}(1-p)^{t-1-k}$
- ◯ $\binom{n}{k} p^{k}(1-p)^{n-k}$
- ◯ $(1-p)^{k-1}$
- ◯ $(1-p)^{t-1-k}$
- ◯ p^{k-1}
- ◯ p^{t-1-k}
- ◯ 1
- ◯ None of the above

(v) [2 pts] What is the expected minimum number of given variables to guarantee that Pacman's position at time $t+1$ is independent of Ghost Z's position at time $t-1$, with dotted edge probability $p < 1$?

- ◯ 1
- ◯ $p^2 + 2p(1-p) + (1-p)^2$
- ◯ 2
- ◯ $2p + 3(1-p)$
- ◯ $2(1-p) + 3p$
- ◯ $p^2 + 4p(1-p) + 3(1-p)^2$
- ◯ 3
- ◯ $3p + 4(1-p)$
- ◯ $3(1-p) + 4p$
- ◯ $2p^2 + 6p(1-p) + 4(1-p)^2$
- ◯ 4
- ◯ None of the above

(c) You want the joint probability of all three agent positions during some timestep t after knowing their positions in the previous timestep, $t-1$.

(i) [2 pts] For this part, let the dotted edge probability $p = 0$. Which of the following are equivalent to $P\left(X_t, Y_t, Z_t | X_{t-1}, Y_{t-1}, Z_{t-1}\right)$?

- ☐ $P(X_t)P(Y_t|X_t)P(Z_t|X_t, Y_t)$
- ☐ $P(Z_t)P(Y_t|Z_t)P(X_t|Y_t, Z_t)$
- ☐ $P(Z_t|X_{t-1}, Y_{t-1}, Z_{t-1})P(Y_t|X_{t-1}, Y_{t-1}, Z_{t-1}, Z_t)P(X_t|X_{t-1}, Y_{t-1}, Z_{t-1}, Y_t, Z_t)$
- ☐ $P(Z_t|X_{t-1}, Y_{t-1}, Z_{t-1})P(Y_t|Y_{t-1}, Z_{t-1}, Z_t)P(X_t|X_{t-1}, Y_{t-1}, Y_t)$
- ☐ $P(Z_t|Z_{t-1})P(Y_t|Y_{t-1}, Z_{t-1}, Z_t)P(X_t|X_{t-1}, Y_{t-1}, Y_t)$
- ◯ None of the above

(ii) [2 pts] Which of the following are equivalent to $P\left(X_t, Y_t, Z_t | X_{t-1}, Y_{t-1}, Z_{t-1}\right)$ for any probability of the dotted edge $p < 1$?

- ☐ $P(X_t)P(Y_t|X_t)P(Z_t|X_t, Y_t)$
- ☐ $P(Z_t)P(Y_t|Z_t)P(X_t|Y_t, Z_t)$
- ☐ $P(Z_t|Z_{t-1})P(Y_t|Y_{t-1}, Z_{t-1}, Z_t)P(X_t|X_{t-1}, Y_{t-1}, Y_t)$
- ☐ $P(Z_t|X_{t-1}, Z_{t-1})P(Y_t|Y_{t-1}, Z_{t-1}, Z_t)P(X_t|X_{t-1}, Y_{t-1}, Y_t)$
- ☐ $\left[p * P(Z_t|X_{t-1}, Z_{t-1}) + (1-p) * P(Z_t|Z_{t-1})\right] P(Y_t|Y_{t-1}, Z_{t-1}, Z_t)P(X_t|X_{t-1}, Y_{t-1}, Y_t)$

○ None of the above

SID: _____

Q3. [11 pts] Tom and Jerry, Continued

Tom and Jerry are playing a game. Each of them has three cards, Rock (R), Paper (Pa), and Scissors (S). The usual rule for Rock-Paper-Scissors applies: R beats S, S beats Pa, and Pa beats R.

Each round, both Tom and Jerry play a card. If Tom's card beats Jerry's card, Tom gets a reward of 1; If there is a tie, Tom gets a reward of 0.5; otherwise Tom gets 0 reward. The played cards cannot be played again. The game ends after three rounds, when neither Tom nor Jerry has any card to play.

However, the game is asymmetric in that Jerry plays according to a fixed pre-made plan regardless of what cards Tom plays. Tom is also aware of the fact that Jerry is playing according to a fixed plan for every game.

(a) Tom has an initial policy π, which is to play card R first, then card Pa, and finally card S.

(i) [3 pts] Tom does not know Jerry's plan, but he wants to evaluate the expected reward he can get with the initial policy by playing with Jerry. Which of the following methods can Tom use to achieve this?

☐ Value Iteration ☐ Policy Iteration ☐ Direct Evaluation
☐ Temporal-Difference Learning ☐ Q-learning ☐ Not Possible

(ii) [2 pts] Tom's friend, Spike, intercepts Jerry's plan and tells it to Tom. Tom knows that Jerry is going to play card S with probability 0.5 and card R with probability 0.5 in the first round, and play one of the remaining cards uniformly at random in the second round.

What is the expected reward of Tom's policy? If there is not enough information, write `NO`.

(iii) [1 pt] Tom learned an optimal policy π^* against Jerry's plan with one of the methods from the previous part. Playing under policy π^* is deterministic, i.e., Tom plays some card with probability 1 conditioned on the cards Jerry previously played in each round. Spike claims that since Jerry uses a stochastic policy, it may be sub-optimal for Tom to only consider deterministic policies. Is Spike correct?

○ Yes, since there are many more stochastic policies than deterministic policies, so considering stochastic policies gives more flexibility.
○ Yes, but not for the reason above.
○ No, because for any plan that Jerry makes, there always exists a deterministic policy that is at least at good as any other policy against Jerry's plan.
○ No, but not for the reason above.

(b) [2 pts] Jerry changed his plan. Spike somehow knew that Jerry changed the plan and told this to Tom. However, Tom has no information about what the new plan is.
Which of the following methods can Tom use to learn an optimal policy against Jerry's new plan? Tom is allowed to play as many times as desired with Jerry to learn the policy.

☐ Value Iteration ☐ Policy Iteration ☐ Direct Evaluation
☐ Temporal-Difference Learning ☐ Q-learning ☐ Not Possible

(c) **(i)** [1 pt] True or False: No matter what Jerry's plan is, there always exists a policy where Tom gets an expected total reward of at least 1.5 with this policy.

○ True ○ False

(ii) [2 pts] Jerry wants to come up with a plan that minimizes Tom's reward under Tom's optimal policy against the plan. What is such an optimal plan for Jerry? Fill in the probability with which Jerry will play each card in the first round.

$P(R) =$ $P(Pa) =$ $P(S) =$

Q4. [19 pts] Mesut-Bot Going to Class

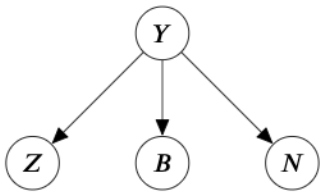
Your friend Mesut-Bot is a very diligent robot and tries to attend all the live lectures and discussions even though there are recordings. However, Pacman thinks it would be fun to make Mesut-Bot accidentally pop into random Zoom rooms, so he mischievously adds a list of random links to Mesut-Bot's calendar. Being a diligent robot himself, Mesut-Bot has hand-picked some features and modeled them as a Naive Bayes classification problem. The features are:

- Z - If the link contains "zoom.us" in it
- B - If the link contains "berkeley" in it
- N - If the link ends with a number

To help your friend Mesut-Bot, you offer him some training data based on your experience in Zoom University. Below is the data and the Naive Bayes structure:

Data	Link	Valid?
1	https://berkeley.zoom.us/j/000000	1
2	https://stanford.zoom.us/j/notaroom	1
3	https://berkeley.zoom.us/j/nowhere	0
4	https://www.twitch.tv/888	1

Data	Link	Valid?
5	https://berkeley.zoom.us/j/11111111	0
6	https://berkeley.zoom.us/j/myroom	1



Knowing that these links are all he has, Mesut-Bot decides to use data $\{1, 2, 3, 4\}$ for training and $\{5, 6\}$ for testing.

(a) [2 pts] Using Maximum Likelihood Estimation without Laplace smoothing, calculate the following values using the training data. If necessary, round your answer to the nearest hundredth (i.e.: 1.2345 would round to 1.23):

- $P(Z = true|Y = 1) =$ _____
- $P(B = false|Y = 0) =$ _____
- $P(N = true|Y = 1) =$ _____

(b) Now Mesut-Bot evaluates the model using data $\{5, 6\}$. The CPTs constructed using Maximum Likelihood Estimation after training are shown below; use these along with your answers from the previous part to calculate the following probabilities. Note that values you were asked to compute in the previous part are omitted from the CPTs (marked as '-' in the tables).

Y	P(Y)
1	0.75
0	0.25

Z	Y	P(Z|Y)
true	1	–
false	1	–
true	0	1
false	0	0

B	Y	P(B|Y)
true	1	1/3
false	1	2/3
true	0	–
false	0	–

N	Y	P(N|Y)
true	1	–
false	1	–
true	0	0
false	0	1

(i) [2 pts] Calculate the following probabilities while evaluating the model for each datapoint. If necessary, round your answer to the nearest hundredth (i.e.: 1.2345 would round to 1.23).

- **Data 5:**
  $P(Y = 1|Z = true, B = true, N = true) =$ _____
  $P(Y = 0|Z = true, B = true, N = true) =$ _____
- **Data 6:**
  $P(Y = 1|Z = true, B = true, N = false) =$ _____
  $P(Y = 0|Z = true, B = true, N = false) =$ _____

(ii) [1 pt] Using your answers from the previous subpart, what are the predictions for each datapoint?

- **Data 5:** ◯ $\hat{Y} = 0$ ◯ $\hat{Y} = 1$
- **Data 6:** ◯ $\hat{Y} = 0$ ◯ $\hat{Y} = 1$

(iii) [1 pt] What is the test accuracy of this model? If necessary, round your answer to the nearest hundredth (i.e.: 1.2345 would round to 1.23)._____

(c) [2 pts] Mesut-Bot is sad that his Naive Bayes model performs so poorly, so he is thinking about ways to improve it. What are the ways that could potentially help Mesut-Bot mitigate bad effects due to **overfitting**?

- ☐ Increase the amount of training data
- ☐ Increase the amount of test data
- ☐ Adding a single feature
- ☐ Use Laplace Smoothing
- ☐ Use Logistic Regression
- ☐ Use a Neural Network

Luckily, Mesut-Bot finds his way to CS 188 lecture. He is asked to classify the following datasets:

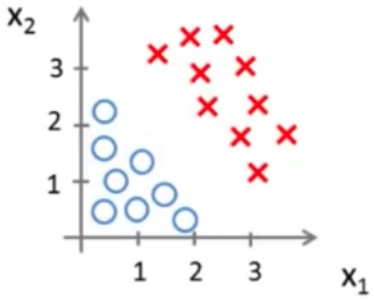


Figure 1: Dataset 1

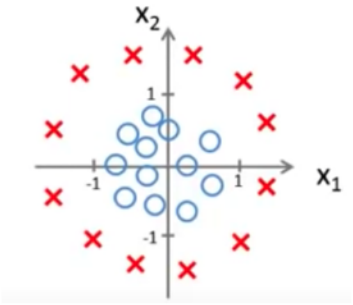


Figure 2: Dataset 2

(d) Using the given features and the specified dataset, which model(s) can achieve a training accuracy of 1?

(i) [1 pt] Dataset 1: X_1, X_2

- ☐ Perceptron
- ☐ Logistic Regression
- ☐ Neural Network
- ◯ None of the above

(ii) [1 pt] Dataset 2: X_1, X_2

- ☐ Perceptron
- ☐ Logistic Regression
- ☐ Neural Network
- ◯ None of the above

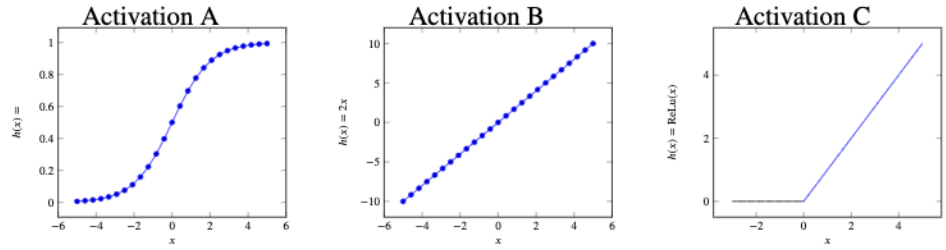
(iii) [1 pt] Dataset 2: X_1, X_2, X_1X_2

- ☐ Perceptron
- ☐ Logistic Regression
- ☐ Neural Network
- ◯ None of the above

(iv) [1 pt] Dataset 2: X_1, X_2, X_1^2, X_2^2

- ☐ Perceptron
- ☐ Logistic Regression
- ☐ Neural Network
- ◯ None of the above

(e) [4 pts] We know that the activation function is an important part of the neural network.



Assuming any network configuration is possible, which of the activation functions above will fit each function with any arbitrary error bound ϵ:

(i) [0 pts] $r(s, a) = \sin \psi$

☐ A ☐ B ☐ C ◯ None

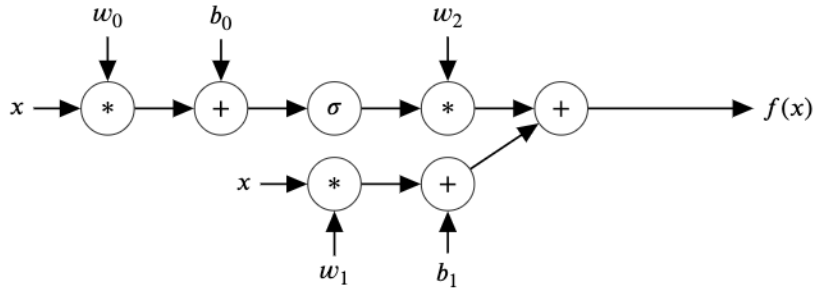
(ii) [0 pts] $f(x) = \begin{cases} 1, & \text{if } x < -1 \\ -x, & \text{if } -1 < x < 0 \\ 0, & \text{otherwise} \end{cases}$

☐ A ☐ B ☐ C ◯ None

(iii) [0 pts] $f(x) = 1$

☐ A ☐ B ☐ C ◯ None

(f) [3 pts] A neural network is presented below. $\sigma(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{e^x+1}$, and $\tanh(x) = \frac{e^{2x}-1}{e^{2x}+1}$.



Which of the following are correct about this neural network?

☐ With sufficient amount of data, this neural network can accurately approximate the function $f(x) = sin(x)$. (You trained a neural network to approximate the $sin(x)$ function in project 5.)

☐ This neural network often generalizes well to unseen data.

☐ A deeper neural network is better at expressing complicated functions than this neural network.

☐ Adding an extra "+" node with bias b_2 before (to the left of) the "$*$" node with coefficient w_1 expands the set of functions this neural network can represent.

◯ None of the above

SID: _____

Q5. [11 pts] Reinforcement Learning on Belief States

In this question, you will be helping reinforcement learning agent Rob to play a simple two-player cooperative game called "Agreed!" against Mesut. The game lasts T rounds in total.

During each round, each player can vote "0" or "1" ($a_t \in [0, 1]$). Both players will get a shared reward of 1 if their votes are the same, and 0 otherwise.

(a) Rob knows that Mesut follows one of these three predetermined policies when deciding his action for a turn:

- π^A = Always vote "0"
- π^B = Always vote "1"
- π^C = Toss a fair (50/50) coin, vote "0" if it lands on tails, and "1" if it lands on heads

Let π_t denote the policy that Mesut is following at round t. Let a_t denote the action performed by Mesut at time t.

Rob is trying to guess what policy Mesut will make next so he calculates, $P(\pi_t = \pi^X \mid a_{0:t}, r_{0:t})$. This table represents Rob's belief of the policy Mesut is following at time t in order to produce the trajectory of actions $a_{0:t}$, while receiving rewards $r_{0:t}$.

Assume a uniform prior for π_0: $P(\pi_0 = \pi^A) = \frac{1}{3}$, $P(\pi_0 = \pi^B) = \frac{1}{3}$, and $P(\pi_0 = \pi^C) = \frac{1}{3}$.

(i) [1 pt] At the initial round (round 0), which of the following are true?

- ☐ $P(\pi_0 = \pi^A \mid a_{0:0} = [0], r_{0:0} = [0]) = 0$
- ☐ $P(\pi_0 = \pi^C \mid a_{0:0} = [0], r_{0:0} = [0]) > 0$
- ☐ $P(\pi_0 = \pi^C \mid a_{0:0} = [1], r_{0:0} = [0]) = 0$
- ☐ $P(\pi_0 = \pi^B \mid a_{0:0} = [1], r_{0:0} = [1]) = 1$
- ◯ None of the above

(b) Mesut's strategy shifts overtime, especially because of the outcome of the current round. Mesut's close friend gives Rob the following transition matrices based on what she knows about Mesut:

Figure 3: Transition matrices for policies, given rewards of the current round

(i) [2 pts] Suppose that Rob follows the **fixed strategy** π^A of always vote "0" regardless of the outcome of each game. Does the Mesut's strategy over time satisfy the Markov property $P(\pi_{t+1} \mid \pi_t) = P(\pi_{t+1} \mid \pi_1, \pi_2, ..., \pi_t)$)?:

- ◯ Yes, because Mesut will not change their policy as long as they keep getting +1 rewards
- ◯ Yes, but not because of the reason above.
- ◯ No, because Mesut's strategy at round $t + 1$ will be affected by rewards received at rounds 0 to t
- ◯ No, but not because of the reason above.

(ii) [2 pts] Now suppose that Rob's policy changes over time among (π^A, π^B, and π^C) based on the same transition matrices above (just like Mesut's). Does your answer above change?

◯ Yes, because when Rob's policy changes over time, Mesut's policy transition can no longer be captured by the two transition matrices.

◯ Yes, but not because of the reason above.

◯ No, because Rob's policy has no impact on $P(\pi_{t+1} \mid \pi_t)$

◯ No, but not because of the reason above.

(iii) [2 pts] After two rounds (round 1), which of the following are true:

☐ $P(\pi_1 = \pi^A \mid a_{0:1} = [0,0], r_{0:1} = [0,0]) > 0$

☐ $P(\pi_1 = \pi^A \mid a_{0:1} = [0,0], r_{0:1} = [0,1]) = 1$

☐ $P(\pi_1 = \pi^B \mid a_{0:1} = [0,0], r_{0:1} = [0,1]) = 0$

☐ $P(\pi_1 = \pi^C \mid a_{0:1} = [1,0], r_{0:1} = [1,0]) > 0$

☐ $P(\pi_1 = \pi^C \mid a_{0:1} = [1,0], r_{0:1} = [1,0]) = 1$

☐ $P(\pi_1 = \pi^B \mid a_{0:1} = [1,0], r_{0:1} = [0,1]) > 0$

☐ $P(\pi_1 = \pi^C \mid a_{0:1} = [1,0], r_{0:1} = [0,1]) > 0$

◯ None of the above

(c) Suppose we deploy our reinforcement learning agent Rob into the real world to interact with Mesut to perform online reinforcement learning. Now we need to decide on a state-space representation. There are two choices:

- Previous observation: $o_t = [a_{t-1}, r_{t-1}]$
- Belief state $b_t = [P(\pi_{t-1} = \pi^A), P(\pi_{t-1} = \pi^B), P(\pi_{t-1} = \pi^C)]$

(i) [1 pt] The size of the state space for representation o_t is _____.

◯ 2 ◯ 3 ◯ 4 ◯ None of the above

(ii) [1 pt] The size of the state space for representation b_t is _____.

◯ 3 ◯ 6 ◯ 9 ◯ 27 ◯ None of the above

(iii) [2 pts] Select all true statements.

☐ Using less expressive representations could be better if we have limited opportunity to deploy Rob to interact with real human

☐ We can perform approximate Q-learning by feeding b_t as the features f_0, f_1, f_2

☐ The optimality of Rob learning with representation b_t does not rely on the accuracy of the transition matrices in part (b)

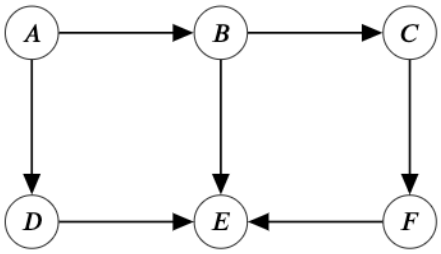
◯ None of the above

SID: _____

Q6. [10 pts] Pac-mate or Pac-poster

(a) Pacman is on a spaceship with six other friends (A, B, C, D, E, F) when he realizes that two of them are imposters! His goal is to figure out the probability of each of his friends being an imposter, so that he can kick the imposters off of the ship.

He considers the following Bayes Net, where all nodes correspond to binary random variables representing each of his friends being imposters. For example, $A = +a$ corresponds to friend A being an imposter and $A = -a$ corresponds to friend A not being an imposter:



(i) [2 pts] Pacman sees friend E faking a task, so he knows that she is one of the imposters. Now Pacman wants to compute $P(A|+e)$, but he's having trouble deciding what technique he should use to calculate it. He first considers Variable Elimination.

What is the most efficient elimination ordering of nodes that minimizes the size of the largest factor created when computing $P(A|+e)$? Specify your answer as a comma-separated list of nodes. For example, if your answer is to eliminate A and then B, you should write 'A,B'. If multiple orderings are just as efficient, write the one that comes first alphabetically.

(ii) [2 pts] Pacman instead decides to use sampling methods to estimate the probabilities. He runs a simulator in the admin room that generates the following sample while trying to compute $P(A|+e)$: $(+a, +b, -c, -d, -e, +f)$. For which of the following sampling approaches could the simulator have been generating samples?

- ☐ Prior Sampling
- ☐ Rejection Sampling
- ☐ Likelihood Weighting
- ☐ Gibb's Sampling
- ○ None of the Above

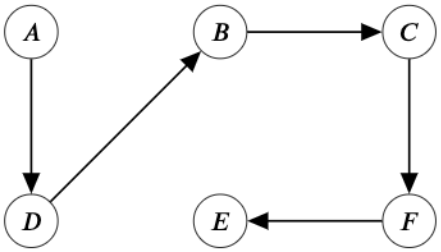
(b) Pacman realizes that there might be something wrong with his Bayes Net, so he tries generating different Bayes Nets instead. Consider each Bayes Net independently, and select the methods that would allow for reasonably efficient computation of $P(A|+e)$ for each Bayes Net. If none are efficient, select 'None of the above'.

Each method is considered efficient if it meets its corresponding criteria below:

- **Variable Elimination:** the largest factor created in the optimal elimination order is smaller than the number of nodes in the Bayes Net.
- **Prior Sampling:** approximately no more than 10% of generated samples can be inconsistent with the evidence.
- **Rejection Sampling:** approximately no more than 20% of samples are rejected.
- **Likelihood Weighting:** the ratio between the largest and the smallest weight that can be assigned is less than $5:1$.

The CPT for E is provided for each Bayes Net. You may assume that every other random variable takes on $+$ with probability 0.5 and $-$ with probability 0.5.

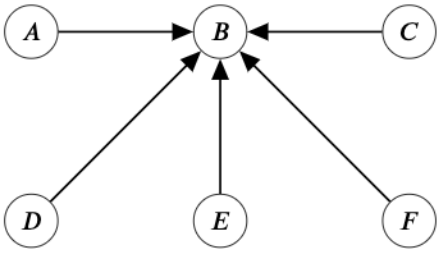
(i) [3 pts]



E	F	$P(E\|F)$
+	+	0.5
+	-	0.5
-	+	0.5
-	-	0.5

- ☐ Variable Elimination
- ☐ Prior Sampling
- ☐ Rejection Sampling
- ☐ Likelihood Weighting
- ○ None of the Above

(ii) [3 pts]



E	$P(E)$
+	0.8
-	0.2

- ☐ Variable Elimination
- ☐ Prior Sampling
- ☐ Rejection Sampling
- ☐ Likelihood Weighting
- ○ None of the Above

SID: _____

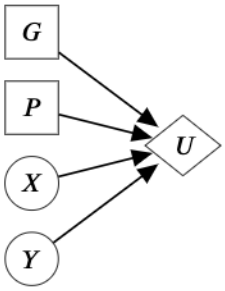
Q7. [17 pts] Value of Pacman Information

For fill in the blank questions, please write in decimals and round to the nearest hundredth. Unsimplified answers or expressions involving variables will not receive credit.

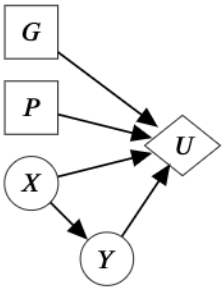
Notation: if $X \sim U(a, b)$, X takes value uniformly on the range $[a, b]$.

(a) [2 pts] Game 1: The ghost chooses a number G and Pacman randomly chooses a number P at the same time. A computer generates a number $X \sim U(0, 10)$ and then another number $Y \sim U(0, X)$. The utility is $f(G, P, X, Y)$ for a fixed function f. Select the decision net(s) that can correctly represent the problem above for the ghost.

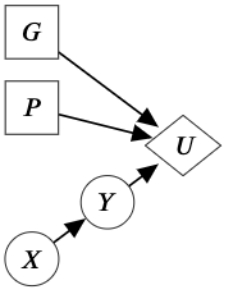
☐ (A)



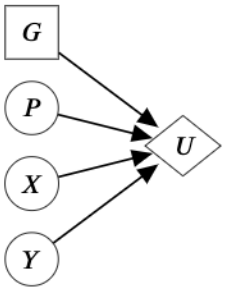
☐ (B)



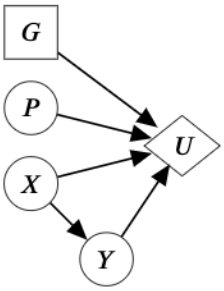
☐ (C)



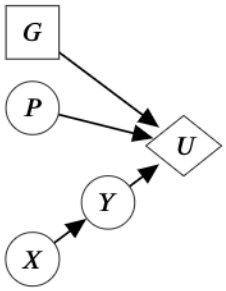
☐ (D)



☐ (E)



☐ (F)



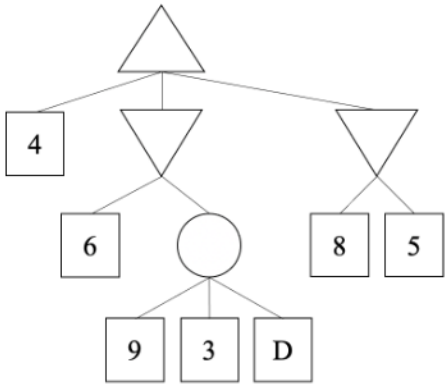
○ (G) None of the above

(b) Game 2: In the game tree below, D is a real number $\sim U(0, 10)$. The ghost does not know the true value of D, but does know that D is sampled from $\sim U(0, 10)$ and wants to achieve the greatest possible score in expectation. Pacman does know the true value of D and uses this information to minimize the ghost's best possible score as much as he can.

Note, the circle node is a random chance node where all 3 option are equally likely.

(i) [5 pts]

Calculate and fill in the values below.



MEU(∅) = ☐

MEU(D) = ☐

VPI(D) = ☐

(c) Game 3: Pacbaby doesn't know how to play games and chooses actions randomly. Pacman has to leave and Pacbaby is taking his place in the game. But the ghost doesn't know! She still believes that she is playing against Pacman, who minimizes her utility (her score in the game).

Note, the Ghost now knows all the leaf node values and the structure of the tree like a normal minimax agent. She believes that Pacman is playing with this information as well.

(i) [1 pt] Select all correct choices from below.

- ☐ The ghost is guaranteed to take the optimal actions.
- ☐ The ghost is guaranteed to not take the optimal actions.
- ☐ We don't have enough information to tell if the ghost will take the optimal actions.
- ○ None of the above.

(ii) [1 pt] Select one choice below.

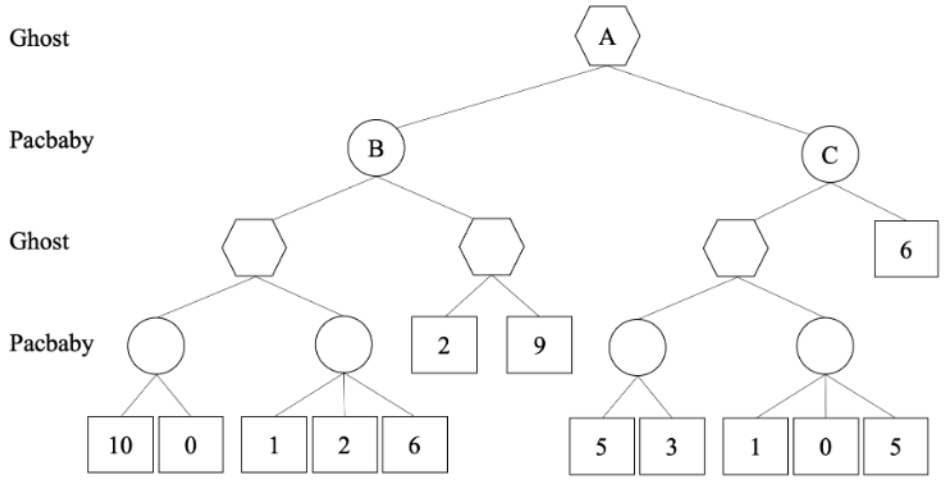
○ Playing against Pacbaby will always give the ghost a score at least as high as the score she would get if she plays the same game against Pacman.

○ Playing against Pacbaby is not guaranteed to give the ghost a score at least as high as the score she would get if she plays the same game against Pacman.

For the remaining parts of this problem, consider the game tree below. Fill in the value of each letter with the score the ghost will receive in expectation if they play out that branch thinking that Pacman is still there playing optimally.

Circle nodes represent Pacbaby's decision which is chosen randomly.

Hexagon nodes represent the Ghost's decision which maximizes the score if Pacbaby were to play as a minimizer.



(iii) [6 pts] $A =$ ☐

$B =$ ☐

$C =$ ☐

(iv) [2 pts] Before the game starts pacbaby cries out and the ghost hears him and realizes she is playing against a random acting player. How much will the ghost's expected score be after adjusting her strategy with this information?

☐