INSTRUCTIONS

- **Due: Monday, March 11, 2024 at 10:00 PM EDT.** Remember that you may use up to 2 slip days for the Written Homework making the last day to submit **Wednesday, March 13, 2024 at 10:00 PM EDT.**
- **Format:** Write your answers in the `yoursolution.tex` file and compile a pdf (preferred) or you can type directly on the blank pdf. Make sure that your answers are within the dedicated regions for each question/part. If you do not follow this format, we may deduct points. We will NOT accept handwritten solutions of any kind.
- **Images:** To insert pictures, we recommend drawing it on PowerPoint or Google Drawings, saving it as an image and including it in your latex source.
- **How to submit:** Submit a pdf with your answers on Gradescope. Log in and click on our class 15-281 and click on the submission titled HW6 and upload your pdf containing your answers. **Misaligned submissions will have at least 5% taken off their score.**
- **Policy:** See the course website for homework policies and Academic Integrity.

Name	
Andrew ID	
Hours to complete?	○ (0, 2] hours ○ (2, 3] hours ○ (3, 4] hours ○ (4, 5] hours ○ (5, 6] hours ○ (6, 7] hours ○ (7, 8] hours ○ > 8 hours

For staff use only

Q1	Q2	Q3	Q4	Q5	Total
/18	/24	/20	/30	/8	/100

Q1. [18 pts] Probability: Product Rule and Bayes Rule

Part 1: Product Rule

Suppose that if we randomly choose a student, the probability that they like to play volleyball is 0.01. Now, suppose that if we randomly choose a student that likes to play volleyball, the probability that they are tall is 0.3. In other words, the probability that a student is tall given that they like to play volleyball is 0.3.

(a) [2 pts] Intuitively, would you expect the probability that a student likes to play volleyball and is tall to be lower or higher than 0.01? (Why?)

◯ Lower

◯ Higher

(b) [8 pts] Consider two binary random variables, L and T. L represents whether you are late for work or not, while T represents whether there's a traffic jam or not. So, $+l, +t$ means that you're late for work and there's a traffic jam. We are given the following probability tables:

T	$P(T)$
$+t$	0.4
$-t$	0.6

L	T	$P(L \mid T)$
$+l$	$+t$	0.8
$-l$	$+t$	0.2
$+l$	$-t$	0.25
$-l$	$-t$	0.75

Compute the four entries of $P(L, T)$.

(L, T)	**P(L, T)**
$(+l, +t)$	
$(+l, -t)$	
$(-l, +t)$	
$(-l, -t)$	

Part 2: Bayes Rule

The product rule allows us to write the joint distribution of two random variables, A and B, in two different ways:

$$P(A, B) = P(A \mid B)P(B)$$

$$P(A, B) = P(B \mid A)P(A)$$

Setting these equal to each other and moving one of the marginal terms to the other side gives us a derivation of Bayes' rule:

$$P(A \mid B)P(B) = P(B \mid A)P(A)$$

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}$$

Bayes' rule is incredibly useful as it relates $P(A \mid B)$ to $P(B \mid A)$ and allows us to calculate one from the other.

As an example, let's take a look at one variant of what is commonly known as the false positive paradox.

In a population of 1000 people, 2% have a deadly disease. You are administering a test for this disease, which has a false positive rate of 5% (i.e. it tests positive when a person doesn't have the disease 5% of the time) and a false negative rate of 0%.

Let T be a random variable indicating whether or not the person tests positive, and D indicate whether or not the person actually has the disease. We then have the following tables:

D	$P(D)$
$+d$	0.02
$-d$	0.98

T	D	$P(T \mid D)$
$+t$	$+d$	1.0
$-t$	$+d$	0
$+t$	$-d$	0.05
$-t$	$-d$	0.95

(c) [8 pts] Compute the four entries in the $P(D \mid T)$ table:

D	**T**	**(D | T)**
$+d$	$+t$	
$+d$	$-t$	
$-d$	$+t$	
$-d$	$-t$	

Q2. [24 pts] Classical Planning and GraphPlan

Suppose we translate the Valet Parking problem from the previous online homework into a classical planning problem with predicates ClearBehind(car) and ParkedBehind(car1,car2). Feel free to refer to the Valet Parking problem specifics in the online homework (Homework 5 Question 4) as necessary, but note that the specific states in this problem may correspond to the diagram in the previous online homework. We define two operations:

ParkBehind(car1, car2)

- Preconditions:
  - ClearBehind(car1)
  - ClearBehind(car2)
  - ParkedBehind(car1, place)
- Add List:
  - ParkedBehind(car1, car2)
  - ClearBehind(place)
  - ¬ ParkedBehind(car1, place)
  - ¬ ClearBehind(car2)
- Delete List:
  - ClearBehind(car2)
  - ParkedBehind(car1, place)

ParkInNewRow(car)

- Preconditions:
  - ClearBehind(car)
  - ParkedBehind(car, place)
  - ¬ ParkedBehind(car, curb)
- Add List:
  - ParkedBehind(car, curb)
  - ClearBehind(place)
  - ¬ ParkedBehind(car, place)
- Delete List:
  - ParkedBehind(car, place)
  - ¬ ParkedBehind(car, curb)

Recall, linear planning works on one goal until it is completely solved before moving on to the next goal. In contrast, non-linear planning considers all possible sub-goal orderings and handles goal interactions by interleaving. The issue with non-interleaved planning methods such as linear planning is that it will naively pursue one subgoal X after satisfying another subgoal Y, but may perform extra steps or may never accomplish the goal because steps required to accomplish X might undo things in subgoal Y. This issue has been coined the Sussman anomaly.

(a) [8 pts] With the following initial state, identify the solution plans a linear and non-linear planner would return using the operators above. Both linear and nonlinear planners will try goals from left to right.

$$State = ParkedBehind(C, A) \wedge ParkedBehind(A, Curb) \wedge$$

$$ParkedBehind(B, Curb) \wedge ClearBehind(B) \wedge ClearBehind(C)$$

Assume all appropriate negated predicates are also in the knowledge base.

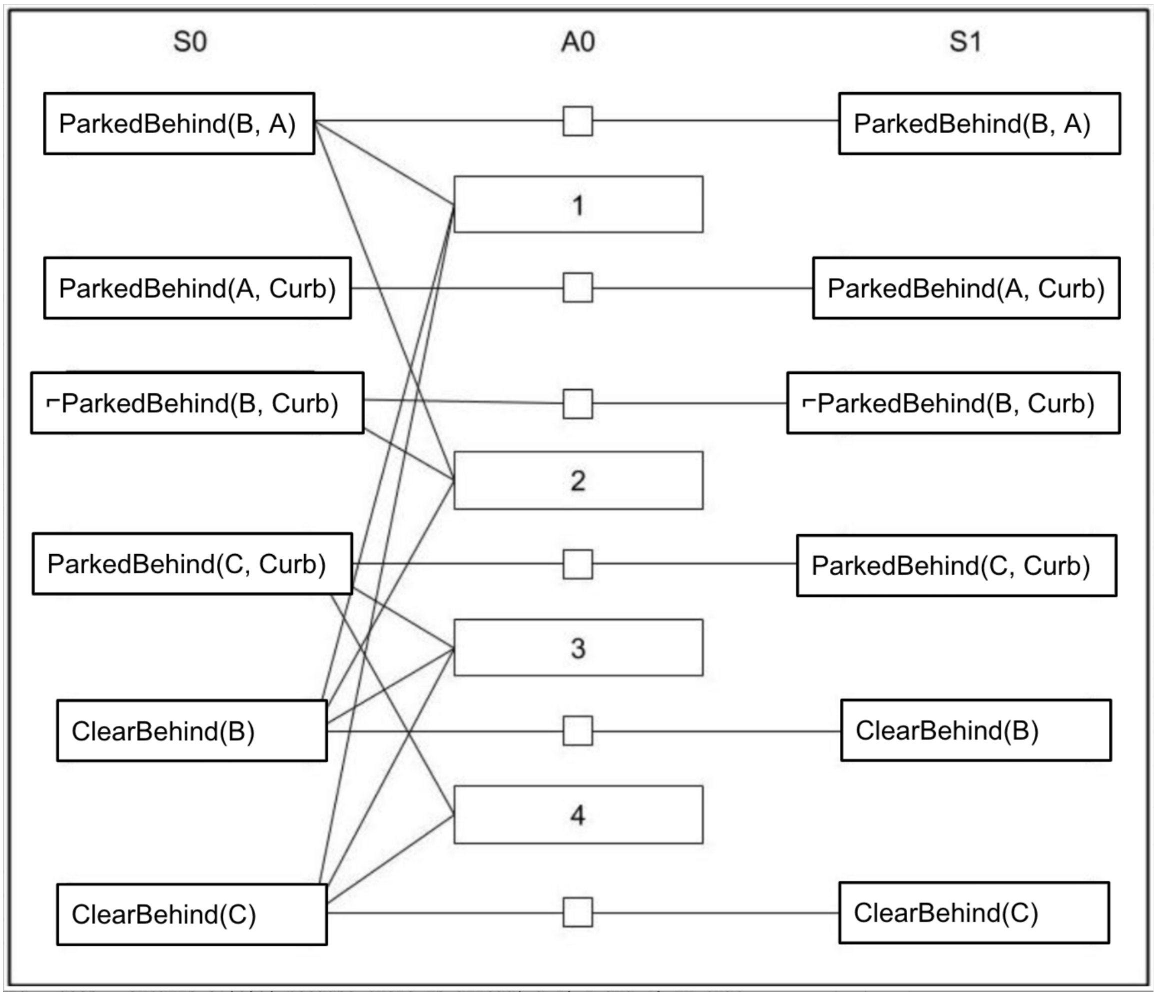
$$Goal = ParkedBehind(A, B) \wedge ParkedBehind(B, C) \wedge ParkedBehind(C, Curb)$$

Linear plan:

Non-linear plan:

(b) [4 pts] Now consider the following image that shows a template for the first two levels of the **GraphPlan graph** for a ValetParking problem. We have drawn in the connections between actions in A0 and their preconditions in S0, as well as persistence actions (unnamed action nodes or **no-ops**). Your task is to:

- Fill in the blanks for the appropriate action nodes in A0 for the boxes labeled 1-4 below.
- Write "N/A" if there is no possible action for the given preconditions. NOTE: normally, when running GraphPlan we won't include such N/A boxes.



1:	**2:**	**3:**	**4:**

(c) [4 pts] Which edges are connected to the state layer S1 as a result of each of the above actions?

- List all the nodes (predicates) in S1 to which there is an ***add*** edge from each of the following actions
- Write "N/A" if the action was not possible
- NOTE: not all predicate nodes are shown in S1 above but you should still include ALL relevant predicates in your response.

1:	**2:**
3:	**4:**

For the following questions, remember that no-op actions count as actions. If you want to use these actions, refer to them as No-op(state) where the precondition and result of No-op(state) is the "state" predicate.

(d) [2 pts] In your completed GraphPlan graph, name two action nodes between which there is an *Inconsistent effects* mutex relation.

Node 1:	**Node 2:**

(e) [2 pts] In your completed GraphPlan graph, name two action nodes between which there is an *Interference* mutex relation.

Node 1:	**Node 2:**

(f) [4 pts] One of the conditions for the GraphPlan algorithm to terminate with a failure is that the graph has **leveled off**. What does this mean? (Choose only one answer)

- ○ A) All possible actions have been explored.
- ○ B) There is no non-empty set of literals between which there are no mutex links.
- ○ C) Two consecutive levels are identical.
- ○ D) The last level of states contains a goal state.

Q3. [20 pts] Planning

Consider a planning environment with six different operations (defined in the table below), starting state A, and goal condition $C \wedge D \wedge E$. Only one operation may be applied at a time, and we are trying to find the plan with the fewest number of operations.

	op1	**op2**	**op3**	**op4**	**op5**	**op6**
Precondition	A	B	A	A	A	A
Add	B	C, D, E	C	D	E	E, ¬A
Delete						

(a) [5 pts]

(i) [3 pts] Run linear planning on this environment with the order of subgoals: C then D then E. What plan is returned?

Plan:

(ii) [1 pt] Is that plan optimal?

◯ Yes ◯ No

(iii) [1 pt] Explain your answer to part (ii).

Answer:

(b) [15 pts]

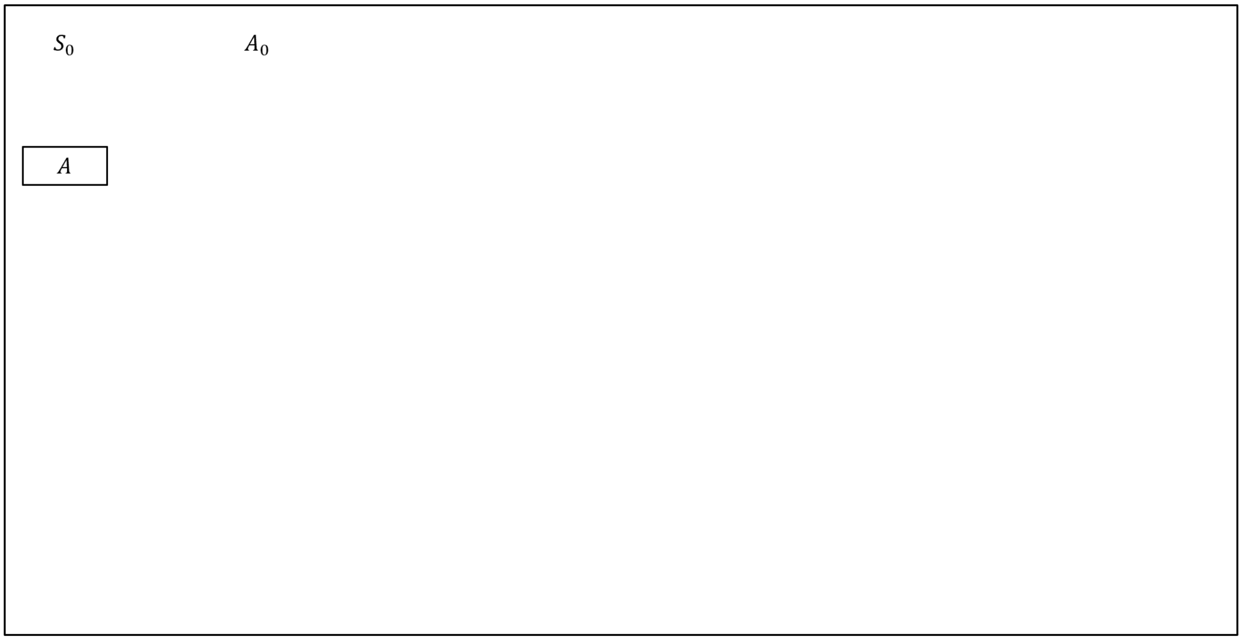
(i) [4 pts] Run GraphPlan on this environment. Draw the **GraphPlan graph**, adding action levels and proposition levels until GraphPlan terminates.

Note: make sure to include the No-op actions for persistent states in your drawing.

For your submission to this problem, you may do one of the following:

- Draw/annotate on top of the existing images in the pdf.
- Edit the `figures/graphplan.png` image file to add markings.

Hand drawing is acceptable, as long as it is clear and precise enough.


S_0
A_0
A

(ii) [3 pts] What plan is returned by GraphPlan?

Plan:

(iii) [2 pts] Is that plan optimal?

◯ Yes ◯ No

(iv) [6 pts] List ALL pairs of exclusive operators in A_0 and ALL pairs of exclusive propositions in S_1. Write 'None' if none exist.

Note: Remember that no-op counts as an action.

Exclusive Operators in A_0:	**Exclusive Propositions in S_1:**

Q4. [30 pts] Machine Learning

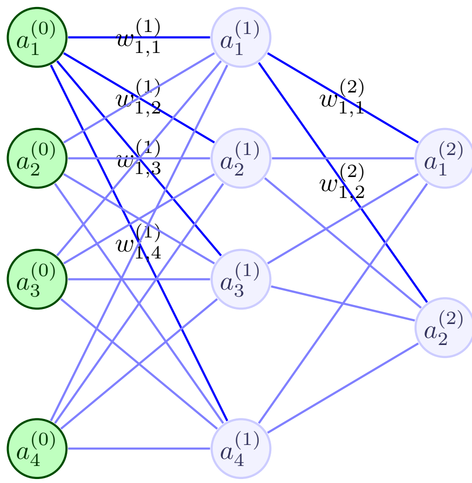
(a) [12 pts] Pinky is trying to predict whether a student will pass 15-281 Midterm 2. Pinky decides to build and train the neural network depicted below, using the following input features for a student:

$a_1^{(0)}$: Midterm 1 exam score

$a_2^{(0)}$: Percentage of lectures attended

$a_3^{(0)}$: Percentage of recitations attended

$a_4^{(0)}$: Average homework score



$$w^{(1)} = \begin{bmatrix} 0.1 & -0.2 & -0.3 & -0.4 \\ 0.1 & 0.2 & -0.3 & -0.4 \\ 0.1 & 0.2 & 0.3 & 0.4 \\ 0.1 & -0.2 & 0.3 & -0.4 \end{bmatrix} \quad w^{(2)} = \begin{bmatrix} 0.1 & 0.2 \\ 0.2 & 0.2 \\ 0.3 & 0.1 \\ 0.1 & 0.3 \end{bmatrix}$$

Note that $w_{i,j}^{(k)}$ refers to the weight corresponding between the connection between the ith neuron in layer $k-1$ and the jth neuron in layer k. After training, Pinky needs your help to predict whether the following student would pass 15-281 Midterm 2:

	Midterm 1 Score	% Lectures Attended	% Recitations Attended	Avg. Homework Score
Student 1	87	92	84	93

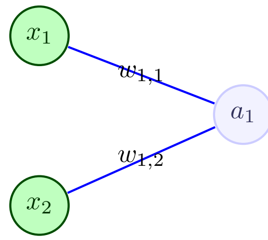
Use the learned weights of the network to find the value before and after the activation at each of the follwing nodes. Assume this network has **no bias terms** and uses a **ReLU activation**, where $\text{ReLU}(x) = \max(0, x)$. *Show your work for partial credit.*

Node	Before Activation	After Activation
Node $a_1^{(1)}$	**i)**	**ii)**
Node $a_2^{(1)}$	**iii)**	**iv)**
Node $a_3^{(1)}$	**v)**	**vi)**
Node $a_4^{(1)}$	**vii)**	**viii)**

(b) Consider the following network architecture.

The network takes in $x_1, x_2 \in \{0, 1\}$, and must output $a_1 \in \{0, 1\}$ for each of the following questions. The network computes the forward process as $a_1 = activation(w^T x + b)$ using **sign** activation

$$sign(x) = \begin{cases} 0 & x \leq 0 \\ 1 & x > 0 \end{cases}$$



(i) [5 pts] Can this network represent the logical **AND** operator between 2 variables x_1, $x_2 \in \{0, 1\}$? If yes, explain why by **specifying** w and b. If not, draw a network by hand or with https://alexlenail.me/NN-SVG/ that can and explain why it works instead.

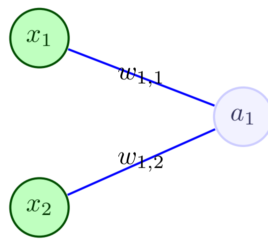
(ii) [5 pts] Can this network represent the logical **OR** operator between 2 variables x_1 and $x_2 \in \{0, 1\}$? If yes, explain why by **specifying** w and b. If not, draw a network by hand or with https://alexlenail.me/NN-SVG/ that can and explain why it works instead.

The following has been reproduced from the previous page for your convenience.

Consider the following network architecture.

The network takes in $x_1, x_2 \in \{0, 1\}$, and must output $a_1 \in \{0, 1\}$ for each of the following questions. The network computes the forward process as $a_1 = activation(w^T x + b)$ using **sign** activation

$$sign(x) = \begin{cases} 0 & x \leq 0 \\ 1 & x > 0 \end{cases}$$



(iii) [8 pts] Can this network represent the logical **XOR** operator between 2 variables x_1 and $x_2 \in \{0, 1\}$? If yes, explain why by **specifying** w and b. If not, draw a network by hand or with https://alexlenail.me/NN-SVG/ that can and explain why it works instead.

Q5. [8 pts] Ethics

Please read the following article and answer the questions below.
`https://www.nytimes.com/2021/11/19/technology/can-a-machine-learn-morality.html`. Note that you can sign up for a New York Times account for free using your andrewID.

(a) [2 pts] Give an example of an AI system **NOT** mentioned by the article where employing morality is important.

Answer:

(b) [2 pts] How do the researchers propose to instill ethical decision-making abilities in machines?

Answer:

(c) [2 pts] What are some potential future implications of machines learning about morality that the article discusses?

Answer:

(d) [2 pts] After reading the article, what are your thoughts on the ethical implications of machines making moral decisions? To what extent should AI agents moral judgments?

Answer: