15-281	Artificial Intelligence	HW5

INSTRUCTIONS

- **Due: Thursday, March 20, 2025 at 11:59 PM EDT.** Remember that you may use up to 2 slip days for the Written Homework making the last day to submit **Saturday, March 22, 2025 at 11:59 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. You may use digital tools (e.g., an iPad) to handwrite your solutions, but make sure they are legible. We reserve the right to take points off if we can't read your solution.
- **How to submit:** Submit a pdf with your answers on Gradescope. Log in and click on our class 15-281, click on the appropriate assignment, and upload your pdf containing your answers.
- **Policy:** See the course website for homework policies and academic integrity.

Name	
Andrew ID	
Hours to complete?	◯ (0, 2] hours ◯ (2, 4] hours ◯ (4, 6] hours ◯ (6, 8] hours ◯ > 8 hours

Q1. [24 pts] Classical Planning and GraphPlan

Suppose we translate the Valet Parking problem into a classical planning problem with predicates ClearBehind(car) and ParkedBehind(car1,car2). We define two operations:

ParkBehind(car1, car2)

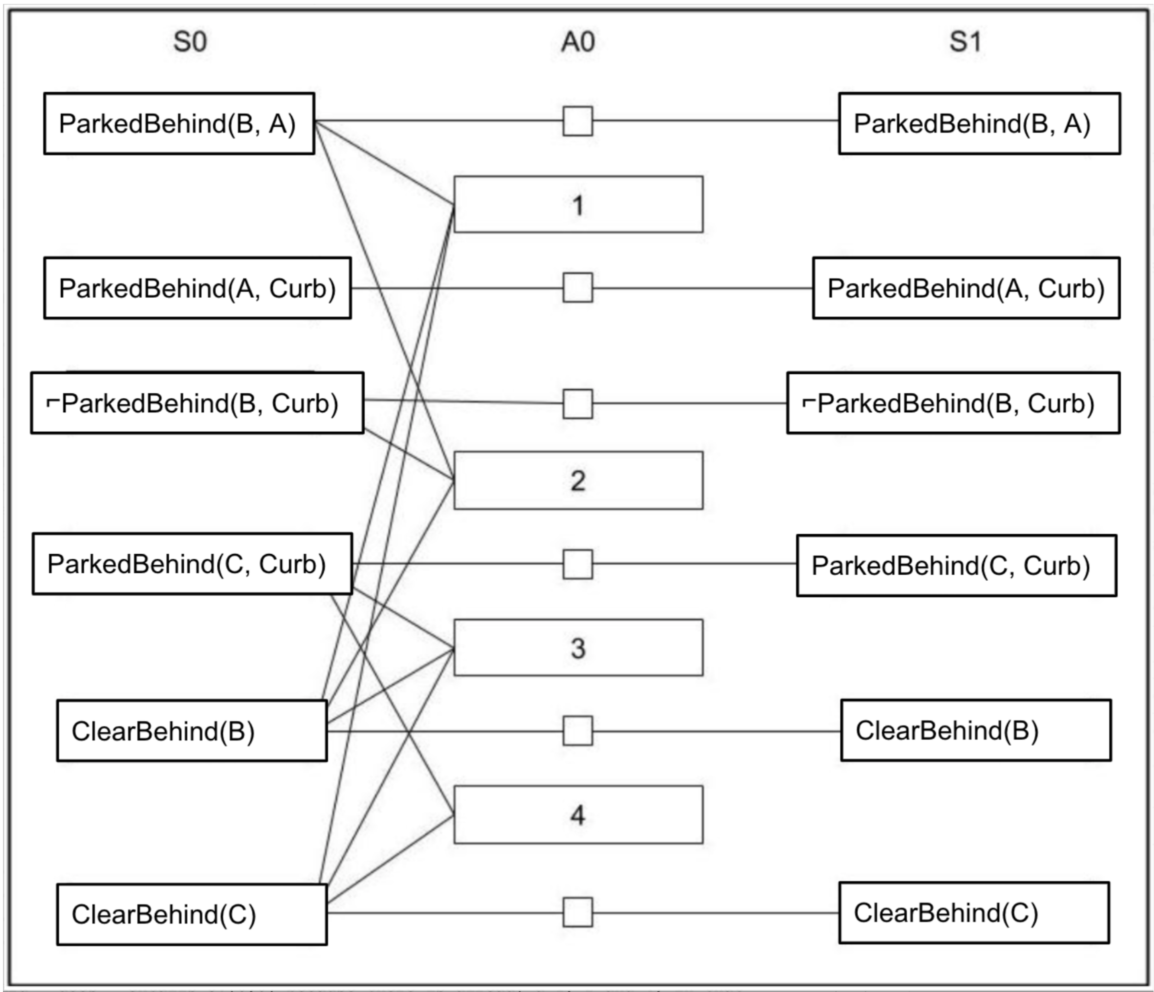
- Preconditions:
  - ClearBehind(car1)
  - ClearBehind(car2)
  - ParkedBehind(car1, place)
- Add List:
  - ParkedBehind(car1, car2)
  - ClearBehind(place)
  - ¬ ParkedBehind(car1, place)
  - ¬ ClearBehind(car2)
- Delete List:
  - ClearBehind(car2)
  - ParkedBehind(car1, place)

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)

(a) [5 pts] 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:

(b) [5 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.

(c) [4 pts] In your completed GraphPlan graph, name two action nodes between which there is an *Inconsistent effects* mutex relation.

Node 1:	**Node 2:**

(d) [4 pts] In your completed GraphPlan graph, name two action nodes between which there is an *Interference* mutex relation.

Node 1:	**Node 2:**

(e) [6 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.

Q2. [22 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						

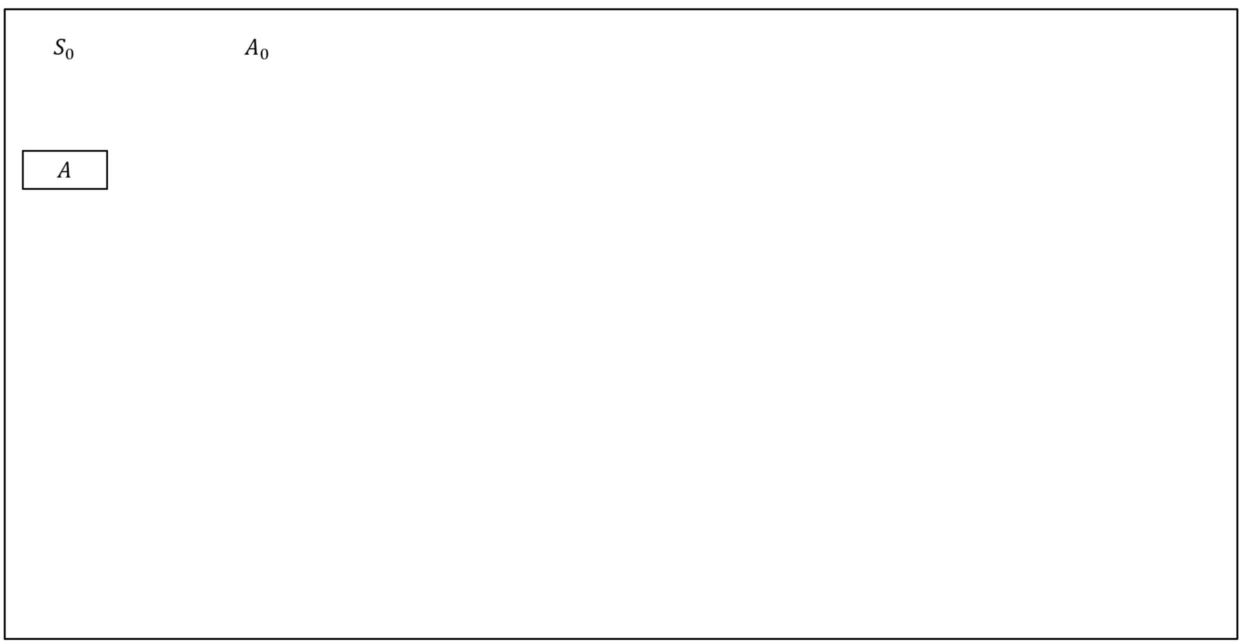
(a) [10 pts] Run GraphPlan on this environment. Draw the **GraphPlan graph**, adding action levels and proposition levels until GraphPlan terminates.

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.



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

Plan:

(c) [3 pts] Is that plan optimal?

◯ Yes ◯ No

(d) [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:**

Q3. [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)**
$+d$	$+t$	
$+d$	$-t$	
$-d$	$+t$	
$-d$	$-t$	

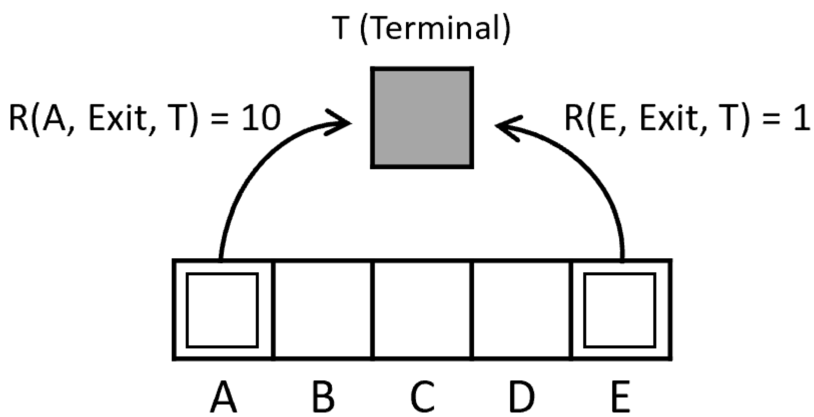
Q4. [22 pts] Solving MDPs

(a) [6 pts]

Please select all that apply:

- ☐ Markov property in MDPs states that action outcomes depend on both the current and past states
- ☐ A key difference between expectimax and value iteration is that expectimax only computes the value for a single state only.
- ☐ The discount factor $\gamma < 1$ suggests that future rewards and current rewards are worth the same.
- ☐ When the discount factor $\gamma = 0$, the agent is acting like a greedy agent.
- ☐ If there is only one action from state s_1, then $Q(s_1, a) = V(s_1)$.
- ☐ None of the above

For parts (b) and (c), consider the Grid World MDP for which Left and Right actions are 100% successful. Specifically, the available actions in each state are to move to the neighboring grid squares. From state A, there is also an exit action available, which results in going to the terminal state and collecting a reward of 10. Similarly, in state E, the reward for the exit action is 1. Exit actions are successful 100% of the time. Exit is the only action available from A and E.



(b) [6 pts] Let the discount factor $\gamma = 0.5$. After running (synchronous) value iteration on this problem, what are the following quantities?

$V_0(D)$	
$V_1(D)$	
$V_2(D)$	
$V_3(D)$	
$V_4(D)$	
$V_5(D)$	

(c) [10 pts] Now let the discount factor $\gamma = 0.2$. Fill in the following quantities.

$V^*(A)$	
$V^*(B)$	
$V^*(C)$	
$V^*(D)$	
$V^*(E)$	

Q5. [14 pts] Machine Learning Design

Ever the diligent student, you decide you need to determine empirically how to improve your grades. After telling all your friends about your idea, you get an email from a suspicious "pinkythepenguin@andrew.cmu.edu" containing a table with the following information, for all 281 students from the past 10 semesters:

1. The number of units the student took that semester
2. The student's GPA before taking the class
3. The student's school (SCS or not SCS)
4. The student's breakfast habits (eats breakfast or does not eat breakfast)
5. **The student's final grade in 281**

Although you are unsure of the ethics of using this mysterious data, you decide to use it to build a model to predict your own 281 grade.

(a) [4 pts]

What type of machine learning problem is this?

1. ◯ Supervised Learning
2. ◯ Unsupervised Learning
3. ◯ Reinforcement Learning
4. ◯ Semi-Supervised Learning

(b) [3 pts] Say you are using a decision tree to model this problem. Describe some reasonable splitting criteria for the decision tree.

Splitting Criteria:

(c) [4 pts] Say you are using a perceptron to model this problem. Describe the weights you would expect the perceptron to learn for each variable. In particular include whether the weights should be positive or negative, and whether they should be large or small. (Multiple answers are possible with justification)

Perceptron Weights:

(d) [3 pts] You show your model to your friends and they laugh at you, saying "An LLM is obviously better than your little perceptron!!" Describe

1. how you would prompt an LLM to solve this problem
2. an ethical concern of using an LLM to solve this problem
3. a new source of data you could include if you used an LLM that you could not (easily) include in either of the previous models

LLM Prompt:
Ethical Concern:
New Data Source: