INSTRUCTIONS

- **Due: Tuesday, October 8, 2019 at 10:00 PM EDT.** Remember that you have NO slip days for Written Homework, but you may turn it in up to 24 hours late with 50% Penalty.
- **Format:** Submit the answer sheet pdf containing your answers. You should solve the questions on this handout (either through a pdf annotator, or by printing, then scanning). Make sure that your answers (typed or handwritten) are within the dedicated regions for each question/part. If you do not follow this format, we may deduct points.
- **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 HW5 and upload your pdf containing your answers.
- **Policy:** See the course website for homework policies and Academic Integrity.

Name	
Andrew ID	
Hours to complete?	

For staff use only

Q1	Q2	Q3	Q4	Q5	Q6	Total
/15	/16	/20	/18	/16	/15	/100

Q1. [15 pts] First-Order Logic

This exercise uses the function predicates In(x,y), Borders(x,y), and Country(x), whose arguments are geographical regions, along with constant symbols for various regions. In each of the following we give an English sentence and a number of candidate logical expressions. For each of the logical expressions, state whether it (1) correctly expresses the English sentence; (2) is syntactically invalid and therefore meaningless; or (3) is syntactically valid but does not express the meaning of the English sentence.

(a) [5 pts] Paris and Marseilles are both in France.

(i) In(Paris $\land$ Marseilles,France).

○ Correct ○ Valid, but incorrect ○ Invalid

(ii) In(Paris,France) $\land$ In(Marseilles,France).

○ Correct ○ Valid, but incorrect ○ Invalid

(iii) In(Paris,France) $\lor$ In(Marseilles,France).

○ Correct ○ Valid, but incorrect ○ Invalid

(b) [5 pts] There is a country that borders both Iraq and Pakistan.

(i) $\exists$ c Country(c) $\land$ Border(c,Iraq) $\land$ Border(c,Pakistan).

○ Correct ○ Valid, but incorrect ○ Invalid

(ii) $\exists$ c Country(c) $\implies$ [Border(c,Iraq) $\land$ Border(c,Pakistan)].

○ Correct ○ Valid, but incorrect ○ Invalid

(iii) [$\exists$ c Country(c)] $\implies$ [Border(c,Iraq) $\land$ Border(c,Pakistan)].

○ Correct ○ Valid, but incorrect ○ Invalid

(iv) $\exists$ c Border(Country(c),Iraq $\land$ Pakistan).

○ Correct ○ Valid, but incorrect ○ Invalid

(c) [5 pts] All countries that border Ecuador are in South America.

(i) $\forall$ c Country (c) $\land$ Border(c,Ecuador) $\implies$ In(c,SouthAmerica).

○ Correct ○ Valid, but incorrect ○ Invalid

(ii) $\forall$ c Country(c) $\implies$ [Border(c,Ecuador) $\implies$ In(c,SouthAmerica)].

○ Correct ○ Valid, but incorrect ○ Invalid

(iii) $\forall$ c [Country(c) $\implies$ Border(c,Ecuador)] $\implies$ In(c,SouthAmerica).

○ Correct ○ Valid, but incorrect ○ Invalid

(iv) $\forall$ c Country (c) $\land$ Border(c,Ecuador) $\land$ In(c,SouthAmerica).

○ Correct ○ Valid, but incorrect ○ Invalid

Q2. [16 pts] General Unifier

For each pair of atomic sentences, give the most general unifier if it exists:

(a) [4 pts] ***P(A, B, B), P(x, y, z)***

θ**:**

(b) [4 pts] ***Q(y, G(A, B)), Q(G(x, x), y)***

θ**:**

(c) [4 pts] ***Older(Father(y), y), Older(Father(x), John)***

θ**:**

(d) [4 pts] ***R(x, y, x, f(y)), R(f(z), z, f(g(2)), w)***

θ**:**

Q3. [20 pts] Classical Planning and GraphPlan

Recall BlocksWorld:

- The robot "hand" can only pick up one block at a time.
- A block is only graspable if there is no block on top of it.
- A block has room for at most one block on top of it.
- The table has unlimited capacity.
- Predicates
  - On(block1, block2)
  - On(block, table)
  - ClearTop(block)

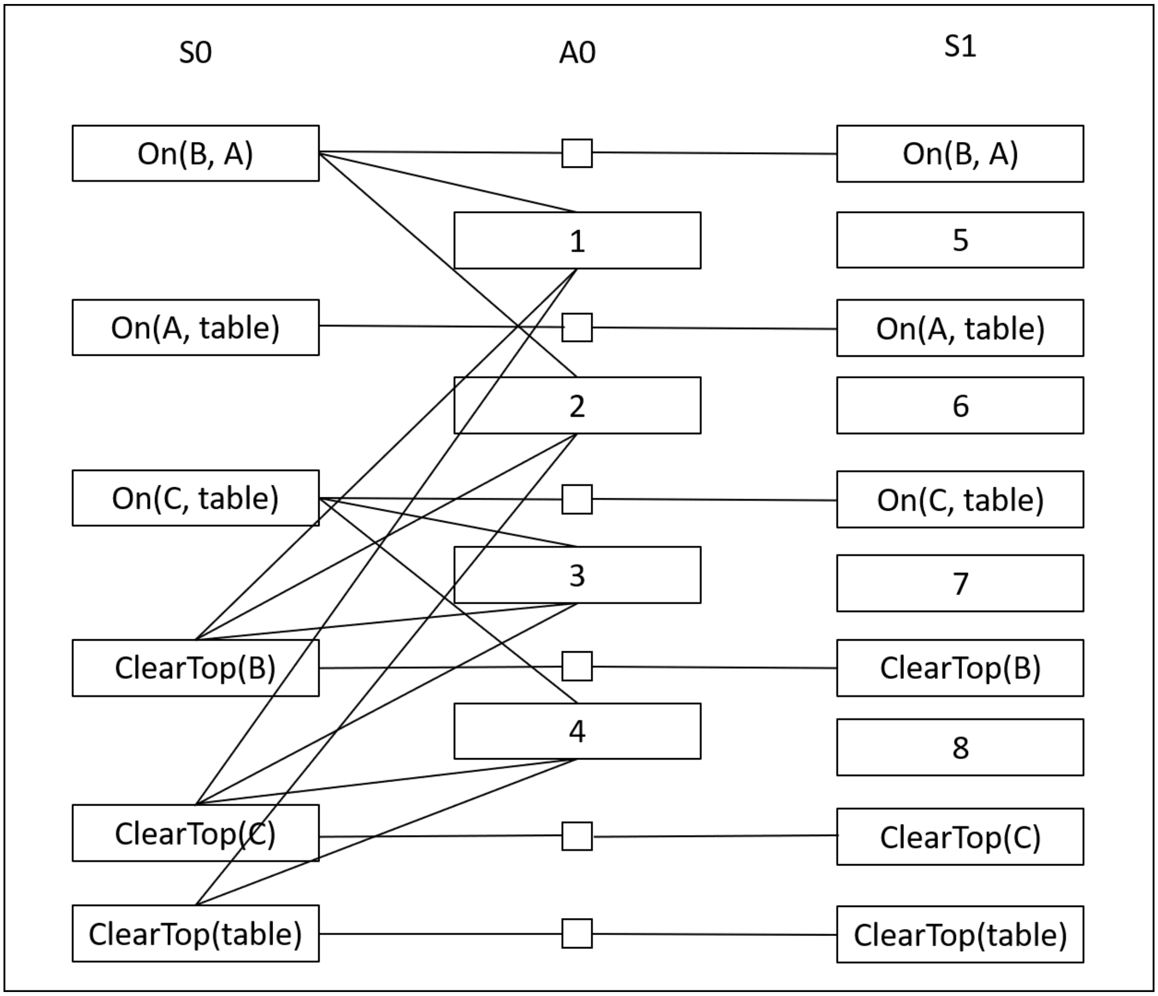
There are two actions in BlocksWorld:

1. MOVE(block, destination)
   - Preconditions:
     - ClearTop(block)
     - ClearTop(destination)
     - On(block, place)
   - Add List:
     - On(block, destination)
     - ClearTop(place)
   - Delete List:
     - On(block, place)
     - ClearTop(destination)
2. MOVETOTABLE(block)
   - Preconditions:
     - ClearTop(block)
     - ClearTop(table)
     - On(block, place)
   - Add List:
     - On(block, table)
     - ClearTop(place)
   - Delete List:
     - On(block, place)

(a) [8 pts] The following image shows a template for the first two levels of the planning graph for a BlocksWorld problem. We have drawn in the connections between actions in A0 and their preconditions in S0. Your task is to:

- Fill in the blanks for the appropriate action nodes in A0 in the boxes below.
- Add any necessary state nodes in S1 in the boxes below.
- **Draw** the edges between action nodes in A0 and state nodes in S1.
  For example, **On(C, table)** → **move(Block, Block)**.

You do not need to explicitly add negated states for actions that delete certain states.



1:	**2:**	**3:**	**4:**
5:	**6:**	**7:**	**8:**

Edge 1:	**Edge 2:**	**Edge 3:**	**Edge 4:**
Edge 5:	**Edge 6:**	**Edge 7:**	**Edge 8:**

For the following section refer to persistence actions (unnamed action nodes) as Persist(state).

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

Node 1:	**Node 2:**

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

Node 1:	**Node 2:**

(d) [3 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 state contains a goal state.

(e) [5 pts] We have discussed two types of planning: linear and non-linear planning. Linear planning works on one goal until it is completely solved before moving on to the next goal. However, 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 fail because steps required to accomplish X might undo things in subgoal Y. This issue has been coined the Sussman anomaly. With the following initial KB, identify the solutions a linear and non-linear planner would return. Both linear and nonlinear planners will try goals from left to right.

$$KB = \{ClearTop(table), On(C, A), On(A, Table), On(B, Table), ClearTop(B), ClearTop(C).\}$$

$$Goal = On(A, B) \wedge On(B, C) \wedge On(C, Table)$$

Linear plan:	**Non-linear plan:**

Q4. [18 pts] Two Mines Company

The Two Mines Company own two different mines that produce an ore which, after being crushed, is graded into three classes: high-, medium- and low-grade. The company has contracted to provide a smelting plant with ***at least*** 12 tons of high-grade, ***at least*** 8 tons of medium-grade and ***at least*** 24 tons of low-grade ore per week. The two mines have different operating characteristics as detailed below. Additionally, we cannot work the same mine for more than 6 days a week.

How many days per week should each mine be operated to fulfill the smelting plant contract?

Note: We have (implicitly) assumed that it is permissible to work in fractions of days.

Mine name	**Cost per day**	**High-grade per day**	**Medium-grade per day**	**Low-grade per day**
Heigh Ho	180	4 tons	5 tons	4 tons
Kessel	160	2 tons	4 tons	8 tons

(a) [8 pts] Write this problem as an LP in **inequality form** as defined in lecture. Define any variables you use in your formulation. *Warning:* Be sure to strictly follow the inequality form, including the proper use of less than or equal, or you will lose points.

Inequality Form:

(b) [8 pts] Accurately plot the graphical representation of this linear program. Specifically:

- Plot the boundary of each halfspace as a line (no need to shade or draw normal vectors), and
- Plot the cost vector as an arrow with magnitude one, somewhere within the feasible region.

Do *not* draw; use a plotting tool such as Python matplotlib and include the resulting image here. Be sure to label the axes of your plot, including tick-marks. Display your plot with a *square* aspect ratio, e.g. in matplotlib: `plt.axis("equal")`. Additionally, zoom your plot to make the entire feasible region visible.

Tip to a plot vector $[v_1, v_2]^T$ in matplotlib starting at some point (x_1, x_2):

```
plt.quiver(x1, x2, v1, v2, angles="xy", scale_units="xy", scale=1)
```

Tip to properly control scaling using a specific width and height:

```
plt.figure(figsize=(width,height))
```

Plot:

(c) [2 pts] Find the optimal solution to the LP problem. Give the solution as days per week per mine as well as the corresponding cost.

Heigh Ho:	**Kessel:**	**Cost:**

Q5. [16 pts] Graphing LPs

For the inequality form of a linear program, and a given A matrix and $\boldsymbol{b}$ vector,

$$\begin{aligned} \min_{\boldsymbol{x}}.\ & \boldsymbol{c}^T \boldsymbol{x} \\ \text{s.t.}\ & A\boldsymbol{x} \preceq \boldsymbol{b} \end{aligned}$$

For each row i of A and $\boldsymbol{b}$, accurately plot 1) the line $a_{i,1}x_1 + a_{i,2}x_2 = b_i$ and 2) the vector $[a_{i,1}, a_{i,2}]^T$ as an arrow beginning at any point on its respective line.

Tip to a plot vector $[v_1, v_2]^T$ in matplotlib starting at some point (x_1, x_2):

```
plt.quiver(x1, x2, v1, v2, angles="xy", scale_units="xy", scale=1)
```

Tip to properly control scaling using a specific width and height:

```
plt.figure(figsize=(width,height))
```

Do *not* draw; use a plotting tool such as Python matplotlib and include the resulting image here. Be sure to label the axes of your plot, including tick-marks. Display your plot with a *square* aspect ratio, e.g. in matplotlib: `plt.axis("equal")`. Additionally, zoom your plot such that all of the vectors are visible.

(a) [8 pts]

$$A = \begin{bmatrix} 3 & 5 \\ 7 & 6 \\ 12 & 6 \end{bmatrix} \qquad \boldsymbol{b} = \begin{bmatrix} 10 \\ 17 \\ 27 \end{bmatrix}$$

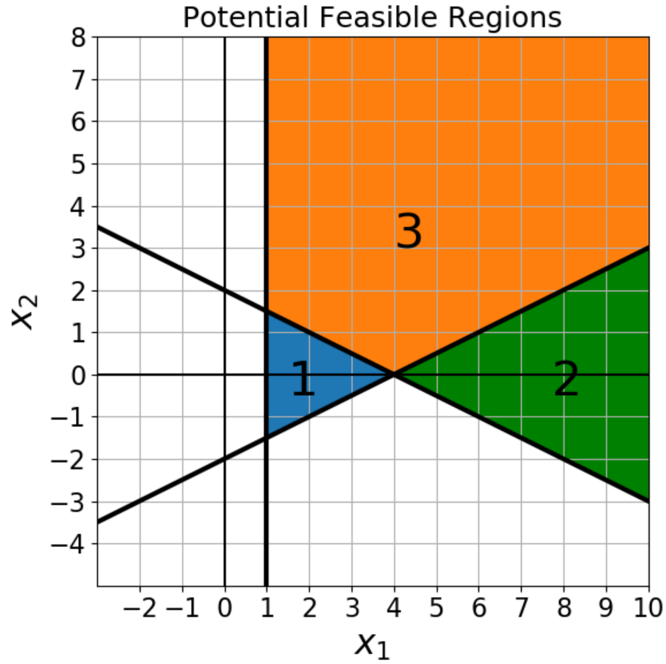
Plot:

(b) [8 pts]

$$A = \begin{bmatrix} -2 & -1 \\ 2 & 5 \\ 7 & 2 \end{bmatrix} \qquad \boldsymbol{b} = \begin{bmatrix} -4 \\ 10 \\ 11 \end{bmatrix}$$

Plot:

Q6. [15 pts] Feasible Regions



In this problem, you are given a graph with constraint boundary lines (**bolded**) and potential feasible regions. You may assume shaded regions at the edge of the plot continue to infinity. Provide the corresponding A and $\boldsymbol{b}$ based on the inequality form below for each feasible region in the boxes below:

$$\min_{\boldsymbol{x}}. \ \boldsymbol{c}^T\boldsymbol{x}$$
$$\text{s.t.} \ A\boldsymbol{x} \preceq \boldsymbol{b}$$

(a) [5 pts] Feasible Region 1

A**:**

$\boldsymbol{b}$**:**

(b) [5 pts] Feasible Region 2

A**:**

$\boldsymbol{b}$**:**

(c) [5 pts] Feasible Region 3

A**:**

$\boldsymbol{b}$**:**