15-281 Artificial Intelligence HW6

INSTRUCTIONS

- **Due: Tuesday, March 3, 2020 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 **Thursday, March 5, 2020 at 10:00 PM EDT.**
- **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 HW6 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
/12	/21	/20	/16	/15	/16	/100

Q1. [12 pts] Resolution

(a) [12 pts] Given the following propositional logic clauses, show E must be true by adding $\neg E$ and using only the resolution inference rule to derive a contradiction. Your answer should be in the form of a graph, where each resolvent is connected by lines to its two parent clauses. Use the clauses below as the initial set of nodes in the graph.

Note: You do not need to use all the nodes, and you may use a node more than once.

$B \vee \neg C$ $A \vee \neg B$ $A \vee \neg D$ $E \vee \neg A \vee \neg B$ C $\neg D$

Q2. [21 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) [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.

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

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

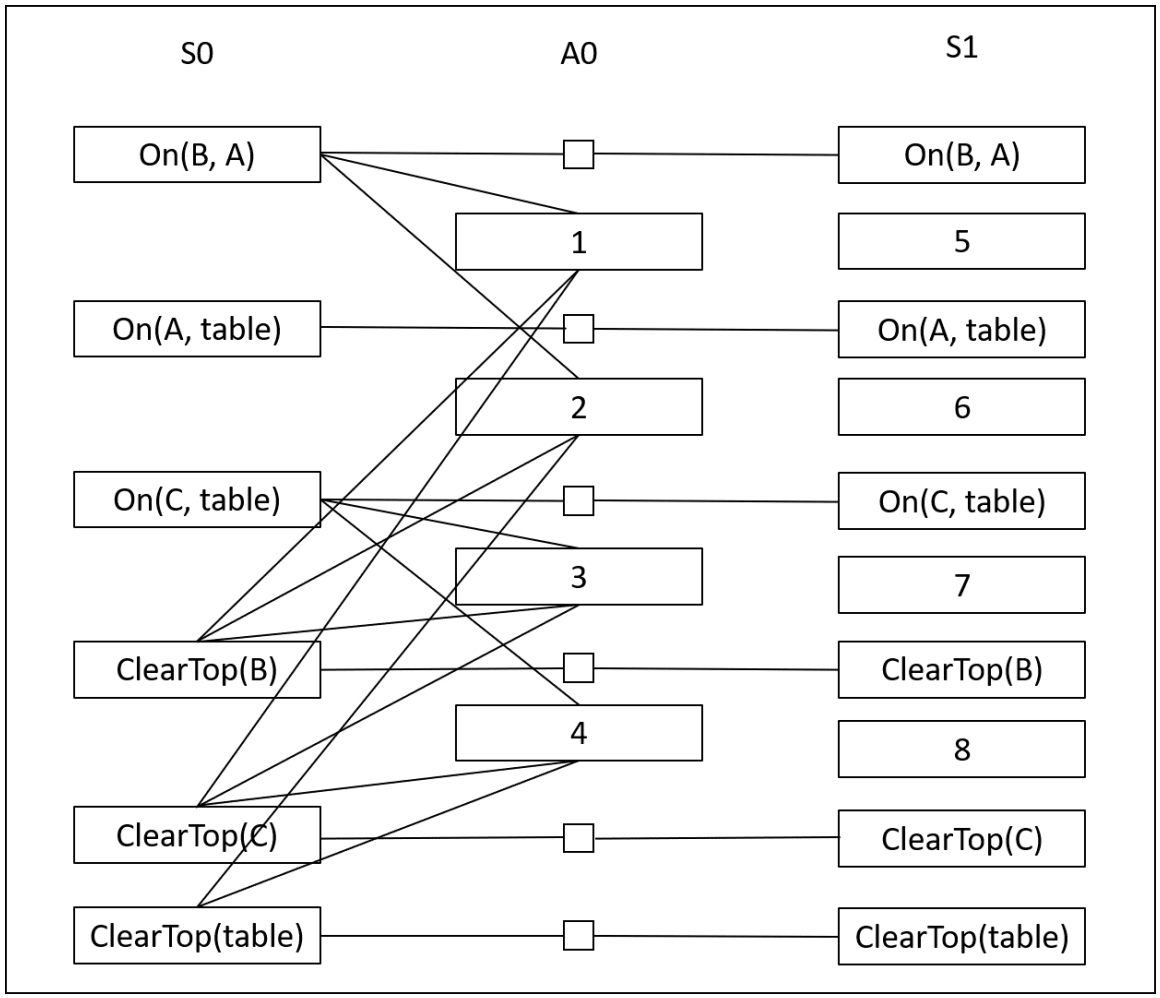
Linear plan:

Non-linear plan:

(b) [8 pts] Now consider the following image shows a template for the first two levels of the **GraphPlan 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.
- Write the edges between action nodes in A0 and state nodes in S1.
  For example, **MoveToTable(Block)** $\rightarrow$ **On(Block, table)**.

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 or **no-ops**) as Persist(state).

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

Node 1:	**Node 2:**

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

Node 1:	**Node 2:**

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

Plan:

Is that plan optimal? ◯ Yes ◯ No

(b) [15 pts] Run GraphPlan on this environment. Draw the **GraphPlan graph**, adding action levels and proposition levels until GraphPlan terminates.

GraphPlan graph:

What plan is returned?

Plan:

Is that plan optimal? ◯ Yes ◯ No

List ALL pairs of exclusive operators in A_0 and ALL pairs of exclusive propositions in S_1. Write 'None' if none exist.

Exclusive pairs:

Q4. [16 pts] Logic

Convert each of the following sentences into predicate logic (or first order logic) using reasonably named predicates, functions, and constants. If you feel a sentence is ambiguous, clarify which meaning youre representing in logic.

(a) [4 pts]

Horses, cows and dolphins are mammals.

(b) [4 pts]

An offspring of a lion is a lion.

(c) [4 pts]

Someone in Pittsburgh likes pierogies.

(d) [4 pts]

All of the houses near Sue's house are either large or old (but not both).

Q5. [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) [3 pts] Paris and Marseilles are both in France.

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

◯ Correct ◯ Valid, but incorrect ◯ Invalid

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

◯ Correct ◯ Valid, but incorrect ◯ Invalid

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

◯ Correct ◯ Valid, but incorrect ◯ Invalid

(b) [6 pts] There is a country that borders both Iraq and Pakistan.

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

◯ Correct ◯ Valid, but incorrect ◯ Invalid

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

◯ Correct ◯ Valid, but incorrect ◯ Invalid

(iii) $\exists$ c Border(Country(c),Iraq $\wedge$ Pakistan).

◯ Correct ◯ Valid, but incorrect ◯ Invalid

(c) [6 pts] All countries that border Ecuador are in South America.

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

◯ Correct ◯ Valid, but incorrect ◯ Invalid

Q6. [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: