AI HW 5
4981113

Problem 1

1.1

1. Object: x = land; c = creatures; t = troll; h, z = humans;
2. Important relation: ContainTrolls(x): a land contains trolls; ContainHumans(x): a land contains humans; track(a,b): a can track b; trap(a,b): a can trap b. willingToPay(a,b): a is willing to pay to b.

There is a land with both trolls and humans:

$\exists x\ ContainsTrolls(x) \wedge \hat{}\, ContainsHumans(x)$

……………………………………………………………………………………..…..1

Troll are very large and noisy creatures:

$\forall c\ troll(c) \Rightarrow large(c) \wedge noisy(c)$ ………..…………………..2

Some of these trolls are aggressive and scare humans:

$\exists t\ aggressive(t) \wedge scareHuman(t)$

………………………………………………………..…………3

Creatures that are larger than humans and aggressive are dangerous

$\forall c\ largerHuman(c) \wedge aggressive(c) \Rightarrow dangerous(c)$

……………………………………………………………………..…4

Thankfully, expert hunters track and trap dangerous creatures for a price.

$\forall h \forall c \exists z\ expertHunters(h) \wedge dangerous(c) \wedge track(h,c) \wedge trap(h,c) \Rightarrow willingToPay(z,h)$

……………………………………………………………………..…5

Devin is an expert hunter

$expertHunter(Devin)$ ..………………………………….…………6

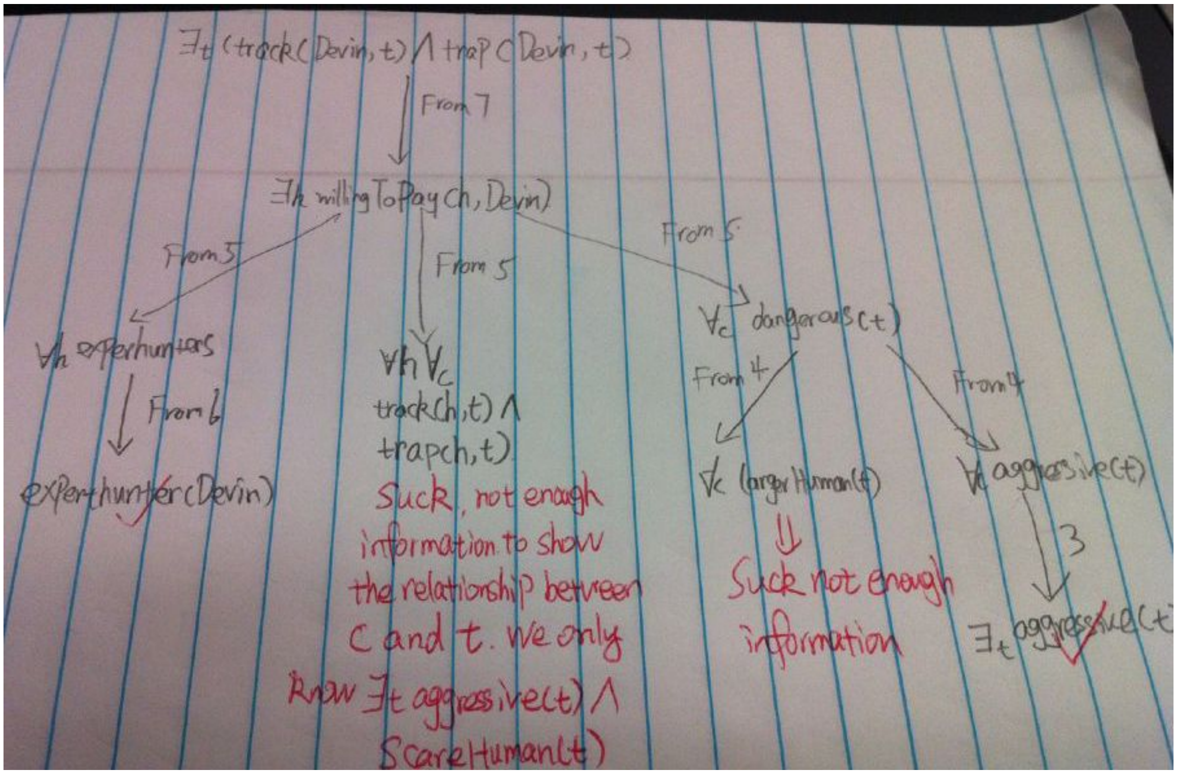
Devin learns of someone who is willing to pay for a troll to be tracked and trapped.

$\exists h \exists t \exists z\ willingToPay(h,z) \Rightarrow track(z,t) \wedge trap(z,t)$ …………….7

1.2.

Devin will track and trap some troll

$\exists t\ (track(Devin, t) \wedge trap(Devin, t))$



Therefore, we can conclude that we can't deduce "Devin will track and trap some troll". The main reason is in the stage 5, we don't have enough information to show the relationship between creatures and trolls even though we have stage2, but we will never reach to stage 2.

Problem 2

2.1

forall x: Youtube(x) → Free(x) ^ (Dog(x) v Cat(x))

$\rightrightarrows$ $\forall x\ (\neg Youtube(x) \vee (Free(x) \wedge (Dog(x) \vee Cat(x)))$

$\rightrightarrows$ $(\neg Youtube(x) \vee (Free(x) \wedge (Dog(x) \vee Cat(x)))$ (drop universal quantifier)

$\rightrightarrows$ $((\neg Youtube(x) \vee (Free(x)) \wedge (\neg Youtube(x) \vee (Dog(x) \vee Cat(x)))$ (And elimination)

$\rightrightarrows$ $((\neg Youtube(x) \vee (Free(x))$ ……………………………………………………1

$(\neg Youtube(x) \vee (Dog(x) \vee Cat(x)))$ ……………………………………….2

$Youtube(SuperSecret)$ ……………………………………………………….3

forall x: Cat(x) → Loss(x)

$\rightrightarrows$ $\neg Cat(x) \lor Loss(x)$ …………………………………………………………...4

forall x: Dog(x) → Loss(x)

$\rightrightarrows$ $\neg Dog(x) \lor Loss(x)$ …………………………………………………………...5

2.2.

Yes. Add $\neg Loss(SuperSecret)$ to KB.

The KB is the following:

$((\neg Youtube(SuperSecret) \lor (Free(SuperSecret))$ ………………………………1

$(\neg Youtube(SuperSecret) \lor (Dog(SuperSecret) \lor Cat(SuperSecret)))$ ………..2

$Youtube(SuperSecret)$ ………………………………………………………….3

$\neg Cat(SuperSecret) \lor Loss(SuperSecret)$ ……………………………..…...4

$\neg Dog(SuperSecret) \lor Loss(SuperSecret)$ ………………………………....5

$\neg Loss(SuperSecret)$ ……………………………………………………...…..6

We can get $(Dog(SuperSecret) \lor Cat(SuperSecret))$ from 2 and 3.

Using $(Dog(SuperSecret) \lor Cat(SuperSecret))$ and 4 can get $(Dog(SuperSecret) \lor Loss(SuperSecret)$.

Finally, using $(Dog(SuperSecret) \lor Loss(SuperSecret)$ and 5, we can conclude $Loss(SuperSecret)$. Contradiction (from 6) reached so the KB entail $Loss(SuperSecret)$

2.3.

Yes. It is possible. Let's assume we add another statement forall x: Loss(x) → Bad(x). Since we add another relation to this KB, the new KB will end with $Bad(SuperSecret)$ with 6. Therefore, we can never reach a contradiction and entail $Loss(SuperSecret)$.

Problem 3

3.1.

Objects: (classroom location, Pen, Seat, Dorm, Paper)

Initial state = At(Dorm) ∧ Holding(Pen)

Goal state = Finish(Exam)

Action:

1. PutInPocket(x)
   Precondition: Holding(x)
   Effect: ReadyToLeave(y)
2. Goto(x ,y)
   Precondition: ReadyToLeave(x) ∧ At(x)

Effect: ¬At(x) ∧ At(y)
3. GetaSeat(x)
Precondition: At(y)
Effect: At(x)
4. TakeOutFromPocket(x)
Precondition: At(y)
Effect: Holding(x)
5. GetExam(x)
Precondition: At(y)
Effect: GetReadyFor(z)
6. Take(x)
Precondition: GetReadyFor(x) ∧ Holding(y)
Effect: Finish(x)

List of actions:
PutInPocket(Pen)
Goto (Dorm, Classroom)
GetaSeat(Seat)
TakeOutFromPocket(Pen)
GetExam(Paper)
Take(Exam)

3.2

Objects: (Classroom location, Pen, Seat, Dorm)
Initial state = At(Dorm) ∧ Holding(Pen)
Goal state = Finish(Exam)
Action:

1. PutInPocket(x)
   Precondition: Holding(x)
   Effect: ReadyToLeave(y)
2. Goto(x,y)
   Precondition: ReadyToLeave(x) ∧ At(x)
   Effect: ¬At(x) ∧ At(y)
3. TakeOutFromPocket(x)
   Precondition: At(y)
   Effect: Holding(x)
4. Take(x)
   Precondition: At(y) ∧ Holding(z)

Effect: Finish(x)

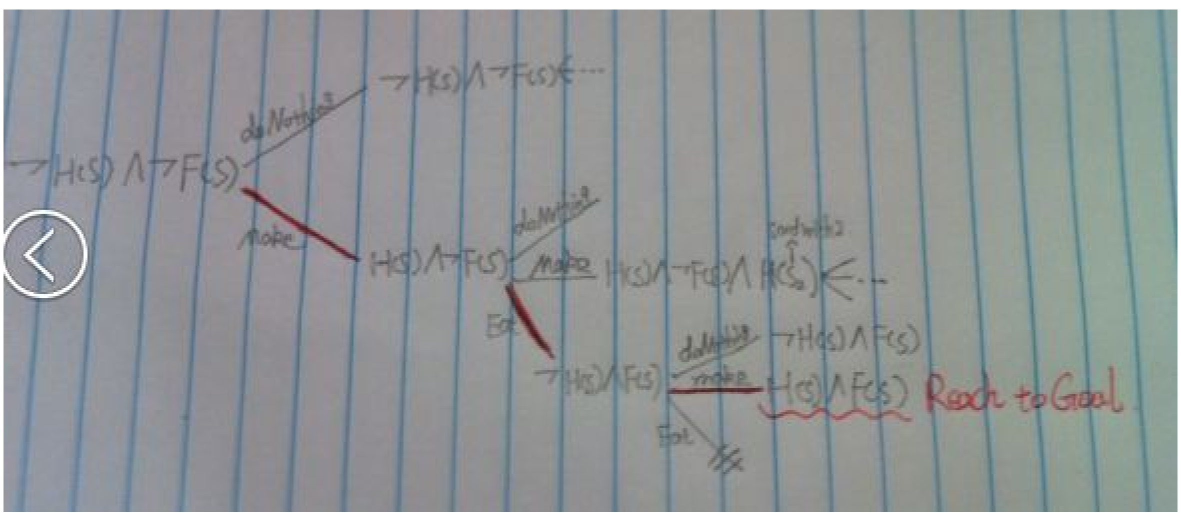
List of actions:
PutInPocket(Pen)
Goto(Dorm,Classroom)
TakeOutFromPocket(Pen)
Take(Exam)

3.3

What we can do is the following: And the first action’s preconditions and effects with the second action’s preconditions and effects. The result of doing this is the same as if you had ran action 1 and action 2 back to back.

Problem 4.

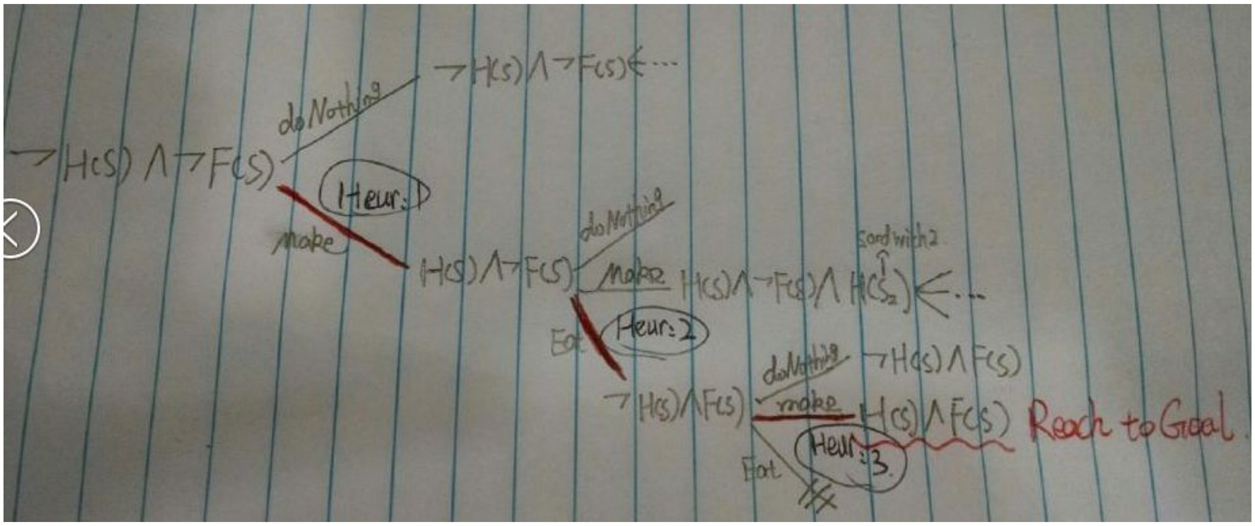
4.1

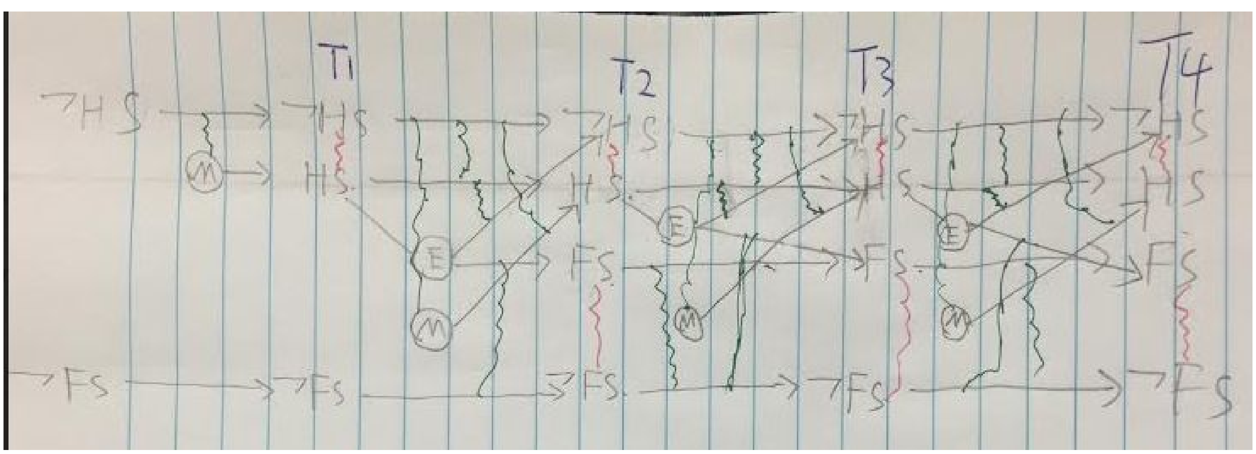


To reach the goal state, we should do Make->Eat->Make

4.2

Since our goal state for this problem doesn’t contain any negative functions/literals, so we should Ignore any deletions in action’ effects to relax this problem.
The heuristic value in this problem I have is the depth distance from the current state to the goal state. I circled my heuristic value in my image below.



Problem 5
5.1


The green lines means there exists some mutexes between actions and the red lines means there exists some mutexes between states.
Compared to level 3 and level 4, we can observe both the mutexes and states no longer change between level 3 and level 4. Therefore, I can conclude this graph-plan converges.

5.2.

Starting from the second level, our goal will be possible. Because the state of H(s) and F(s) don't have any mutexes between them. Moreover, the action Make(x) and Eat(x) also don't contain a mutex since the precondition of Make(x) is nothing and the effect of Eat(x) is F(s). They will not affect with each other. To find the goal, we should start with the goals and then for each stage, propagate all your states and apply the actions. This should lead to even more states. Stop once the state repeat. For the second level, since we can have conflict with state H(s) and F(s) if you are starting from goal, so goal can not be achievable at the second level. Actually, the goal is at the third level.

5.3.

Let's say in a graph-plan problem, there exists two relations/literals with no mutex when the graph is converge. However, I am not sure whether those two literals' precondition will be true simultaneously or not. Therefore, one of state's precondition may false and the other may true, which means only one of the state can be reached, but not all of them. Therefore, pair of relations is impossible to satisfy simultaneously in this case.