# CS575 – Test 1 – Solution

12:30pm – 1:20 am

**1.** (5 point) Translate the following in to a propositional caculus theory: *If prices are high, then wages are high. Prices are high or there are price controls. Also, if there are price controls, then there is not an inflation. However, there is an inflation.*

Is the sentence *Wages are high* entailed by the above theory? (simply answer yes/no will not give you any point)

**Answer:** NOTE: Propositional theory does not have variables, function symbols, predicate symbols

Let $PH$, $WH$, $PC$, and $I$ denote "prices are high", "wages are high", "there are price controls", and "there is an inflation", respectively. Then,

| the theory $T$: | equivalent to | |
|---|---|---|
| $PH \Rightarrow WH$ | $\neg PH \vee WH$ | (1) |
| $PH \vee PC$ | $PH \vee PC$ | (2) |
| $PC \Rightarrow \neg I$ | $\neg PC \vee \neg I$ | (3) |
| $I$ | $I$ | (4) |

(3) + (4) implies $\neg PC$ (5) Using the inference rule: $\frac{\alpha \vee \beta, \neg\beta}{\alpha}$

(5) + (2) implies $PH$ (6)

(6) + (1) implies $WH$

Thus, $T$ entails $WH$, i.e., our theory entails "Wages are high"

**2.** (5 point) Translate the following sentenses into first order logic theory $T$ (you language needs not have function symbol or constant) *Everyone has a parent. The parent of a parent is a grandparent.*

Let $John$ be a person. Prove that your theory $T$ entails that $John$ has a parent.

**Answer:** NOTE: First order theory is NOT a logic program

Use the following predicates:
$person(X)$ - $X$ is a person
$parent(X, Y)$ - $X$ is a parent of $Y$
$has\_parent(X)$ - $X$ has a parent
$is\_grandparent(X)$ - $X$ is a grandparent

The theory $T$ consists of:

$\forall X[person(X) \Rightarrow has\_parent(X)]$
$\forall X \forall Y \forall Z[parent(X, Y) \wedge parent(Y, Z) \Rightarrow is\_grandparent(X)]$.

For $X = john$, using the first sentense in $T$, $\forall X[person(X) \Rightarrow has\_parent(X)]$, we have that $person(john) \Rightarrow has\_parent(john)$ (1) Using the inference rule: $\frac{\forall x P}{P\{x/c\}}$.

$john$ is a person, hence $person(john)$ is true. Together wiht (1) we have that $T \models has\_parent(john)$.

**Another possible solution:**

We can use the following predicates:
$person(X)$ - $X$ is a person
$parent(X, Y)$ - $X$ is a parent of $Y$
$grandparent(X, Y)$ - $X$ is a grandparent of $Y$

Then, the translation would be

$$\forall X[person(X) \Rightarrow \exists Y.[parent(Y, X)]]$$
$$\forall X \forall Y \forall Z[parent(X, Y) \wedge parent(Y, Z) \Rightarrow grandparent(X, Z).$$

Using the first formula and the rule $\frac{\forall x P}{P\{x/c\}}$ and the fact that $person(john)$ is true, we have that $\exists Y parent(Y, john)$ is true. That is, $john$ has a parent.

**3.** (2 point) Attempt to unify the following pairs of expressions. Either show their most general unifiers or explain why they will not unify.

(a) (1 point) $p(X, Z)$ and $p(a, Y)$.

(b) (1 point) $p(X, X)$ and $p(a, b)$.

**Answer:**

(a) $\sigma_0 = \{\}$, $D_0 = \{X, a\}$,
$\sigma_1 = \{X/a\}$, $p(X, Z)\sigma_1 = p(a, Z)$ and $p(a, Y)\sigma_1 = p(a, Y)$; $D_1 = \{Z, Y\}$
$\sigma_2 = \{X/a\}\{Z/Y\} = \{X/a, Z/Y\}$, $p(X, Z)\sigma_2 = p(a, Y)\sigma_2 = p(a, Y)$
A mgu of $p(X, Z)$ and $p(a, Y)$ is $\{X/a, Z/Y\}$.

(b) $\sigma_0 = \{\}$, $D_0 = \{X, a\}$,
$\sigma_1 = \{X/a\}$, $p(X, X)\sigma_1 = p(a, a)$ and $p(a, b)\sigma_1 = p(a, b)$; $D_1 = \{a, b\}$
No variable in $D_1$, the two are not unifiable.

**4.** (4 point) Show that the following program has **only one stable model**.

$$\begin{array}{lcl} p & \leftarrow & not\ q. \\ q & \leftarrow & r, not\ p. \\ p & \leftarrow & not\ p. \end{array}$$

**Answer:** Let the program be $P$. Assume that $S$ is a stable model of $P$. Consider two cases:

**Case 1:** $p \notin S$. Then, because of the rule $p \leftarrow not\ p$, $P^S$ must contain the rule $p \leftarrow$. This means that the minimal model of $P^S$ contain $p$, i.e., $p \in S$. This contradicts with $p \notin S$. So, this case cannot happen.

**Case 2:** $p \in S$. Then, the last two rules of $P$ will be removed when constructing $P^S$. This also means that $q$ and $r$ cannot belong to the minimal model of $P^S$. Thus, the only possibility is

$S = \{p\}$. Indeed, $S = \{p\}$ is a stable model of $P$ because $P^S$ contains only the rule $p \leftarrow$, i.e., $\{p\}$ is equals the minimal model of $P^S$.

**5.** (6 points) Let $\Pi$ be a program that represents the information about courses, students, and a regulation for enrollment into a class that says that a student cannot take a class if she/he had not taken one of its prerequisites.

$$
\begin{array}{rcl}
can\_enroll(S,C) & \leftarrow & student(S), course(C), taken\_all\_pre(S,C). \\
not\_all\_pre(S,C) & \leftarrow & student(S), course(C), prerequisite(C1,C), not\ taken(S,C1). \\
taken\_all\_pre(S,C) & \leftarrow & student(S), course(C), not\ not\_all\_pre(S,C). \\
prerequisite(math279, ai2) & \leftarrow & \\
prerequisite(ai1, ai2) & \leftarrow & \\
student(son) & \leftarrow & \\
student(emi) & \leftarrow & \\
course(ai1) & \leftarrow & \\
course(ai2) & \leftarrow & \\
course(math279) & \leftarrow & \\
taken(emi, math279) & \leftarrow & \\
taken(emi, ai1) & \leftarrow & \\
taken(son, ai1) & \leftarrow &
\end{array}
$$

(a) (1 point) Show that the program is stratified.

(b) (4 point) Compute the stable model of $\Pi$.

(c) (1 point) Is $can\_enroll(son, ai2)$ entailed by $\Pi$? How about $can\_enroll(emi, ai2)$? Why?

**Answer**

(a) We just need to draw the dependency graph of $\Pi$ and there is no cycle with negative link in it, so $\Pi$ is stratified.

(b) Let $S$ be a stable model of $\Pi$. Because of $\Pi$ is stratified, we know that $\Pi$ has only one stable model. Thus, all we need to do is to decide for every atom in $B_P$ whether $a$ belongs to $S$ or not. We observe:

The last ten rules of $\Pi$ are facts, so any stable model $S$ of $\Pi$ must contain these facts. Let us call these facts $X_0$.

Except the facts, $\Pi$ does not contain any rule whose head is one of the predicates $student$, $taken$, $course$, and $prerequisite$. This implies that any atom $a$ in $B_\Pi$ and $a \in S$ and $a$ is constructed from these predicates and terms in $U_\Pi$ then $a$ must belong to $X_0$

The above two observations allow us to conclude that we only need to decide whether atoms of the form $can\_enroll(S,C)$, $taken\_all\_pre(S,C)$, $not\_all\_pre(S,C)$ in $B_\Pi$ belong to $S$.

Because of the second rule, we have that the only atom of the form $not\_all\_pre(X,Y)$ that belongs to $S$ is $not\_all\_pre(son, ai2)$.

This implies that every atom of the form $taken\_all\_pre(S,C)$ except $taken\_all\_pre(son, ai2)$ belong to $S$ because of the third rule.

The above, together with the first rule, allows us to conclude that every atom of the form $can\_enroll(S, C)$ except $can\_enroll(son, ai2)$ belongs to $S$. This concludes the computation of $S$.

(c) From the above, $\Pi \not\models can\_enroll(son, ai2)$ because the only stable model of $\Pi$ does not contain $can\_enroll(son, ai2)$; and $\Pi \models can\_enroll(emi, ai2)$ because $S$ contains $can\_enroll(emi, ai2)$.