CSCE 580: Artificial Intelligence
Written Homework 3: Propositional and First-Order Logic
Due: 2/23/2024 at 11:58pm

You may either type your homework or upload a scanned copy of your written homework. However, your answers must be legible. **Answers that are not legible will be marked as incorrect.**

Turn in **PDF** (not Word or any other format) of your answers to Blackboard.

1 Conjunctive Normal Form in Propositional Logic (20 pts)

What is the correct conversion of the propositional logic statement below to conjunctive normal form? Show your work as you take the following steps: 1) eliminate biconditionals, 2) eliminate implications, 3) push negations inward, 4) distribute disjunctions over conjunctions. You can use a truth table to check your answer.

$(A \wedge B) \leftrightarrow (C \vee \neg A)$

(A ^ B ^ (C v -A)) ^ (-(A ^ B) v (C v -A))

(A ^ B ^ (-A v C)) ^ (-(A ^ B) v (C v -A))

(A ^ B ^ (-A v C)) ^ ((-A ^ -B) v (C v -A))

(A ^ B ^ -A v A ^ B ^ C) ^ (-A v -B v C v -A)

(B ^ -A v B ^ C) ^ (-A v -B v C v -A)

2 Propositional Logic (20 pts)

Your knowledge base contains the following sentences:
$S \rightarrow (K \vee R)$
$W \rightarrow (C \vee R)$
$K \vee C$
$\neg K \vee \neg C$

Prove: $(S \wedge W) \rightarrow R$

2.1 Negation (2 pt)

Negate the sentence that you want prove.

(S ^ W) ^ -R

2.2 Conjunctive Normal Form (9 pts)

Convert all sentences in your knowledge base and the negated sentence that you want to prove to conjunctive normal form. Put **one** clause per line. You may not need to use all lines.

1) (S ^ W) ^ -R

2) -S v -W v R

3) -S v (K v R)

4) -W v (C v R)

5) (-S v K v R)) ^ (-W v (C v R))

6) (-S ^ -W) v ((K v R) ^ (-W v (C v R)))

7) (-S ^ -W) v ((K v R) ^ (-W v K) ^ (-W v C))

8) (-S ^ -W) v (R ^ (-W v K) ^ (-W v C))

9) (-S ^ -W) v R

10) (S ^ W) → R

11) ______

12) ______

2.3 Resolution (9 pts)

Using the sentences in conjunctive normal form, use resolution to do proof by contradiction. Do **one** resolution step per line. You may not need to use all lines. Your resolution steps must be of the form: Resolve W with X to obtain Z.

Resolve (-S ^ -W ^ R) with (-S ^ -W ^ -W v K) to obtain -W ^ R v K.

Resolve (-W ^ R v K) with (-S ^ -W ^ -W v C) to obtain -S ^ R v K v C.

Resolve (-S ^ R v K v C) with (-S ^ -W ^ -W v K) to obtain -W ^ R v K v C.

Resolve -W ^ R v K v C with (S ^ W) → R to obtain -W v K v C.

Now we have a clause that does not contain R. This indicates that our assumption (S ^ W) → R is contradictory.

Therefore, proof by contradiction.

3 English to FOL Conversion (20 pts)

For each English sentence, write the FOL sentence that best expresses its intended meaning. Use *Dog*(x) for "x is a dog", *Bone*(x) for "x is a bone", and *Likes*(x, y) for "x likes y".

Example

"Every dog likes every bone."

$\forall x, y\ Dog(x) \wedge Bone(y) \rightarrow Likes(x, y)$

3.1 (4 pts)

"Some dogs like some bones."

Ex,y Dog(x) ^ Bone(y) → Likes(x, y)

3.2 (4 pts)

"For every dog, there is a bone that the dog likes"

Ax, Ey Dog(x) → Bone(y) ^ Likes(x, y)

3.3 (4 pts)

"For every bone, there is a dog who likes that bone."

Ay, Ex Bone(y) → Dog(x) ^ Likes(x, y)

3.4 (4 pts)

"There is a bone that every dog likes."

Ey, Ax Dog(x) ^ Bone(y) → Likes(x, y)

3.5 (4 pts)

"There is a dog who likes every bone."

Ex, Ay Dog(x) ^ Bone(y) → Likes(x, y)

4 Unification (FOL) (20 pts)

Provide a substitution that provides the **most general unifier** for the two FOL sentences or put None if none exists.

4.1 (5 pts)

P(A,B,B)
P(x,y,z)

{x/A,y/B,z/B}

4.2 (5 pts)

Q(x, G(A,B))
Q(y, G(y,y))

{x/y,G(A,B)/G(y,y)}

4.3 (5 pts)

Neighbor(Penny, Friend(x))
Neighbor(y, Friend(Brother(Sheldon)))

{y/Penny,x/Brother(Sheldon)}

4.4 (5 pts)

Knows(Jack, x)
Knows(y, F(z))

{x/F(z),y/Jack,z/z}

5 First Order Logic (20 pts)

Your knowledge base contains the following sentences in English: All ripe tomatoes are red. There exists a ripe tomato.

Prove: There exists a red tomato.

5.1 Translation (5 pts)

Convert all sentences in the knowledge base and the sentence you want to prove to sentences written in first-order logic. Use the predicates *Tomato*(*x*) for "*x* is a tomato", *Red*(*x*) for "*x* is red", and *Ripe*(*x*) for "*x* is ripe".
Example:
"There exists a red tomato that is not ripe."

$\exists x\, Tomato(x) \wedge Red(x) \wedge \neg Ripe(x)$

All ripe tomatoes are red: Ax (Tomato(x) ^ Ripe(x)) → Red(x)

There exists a ripe tomato: Ex Tomato(x) ^ Ripe(x)

There exists a red tomato: Ex Tomato(x) ^ Red(x)

5.2 Negation (1 pt)

Negate the sentence that you want prove.

Ax (-Tomato(x) v -Red(x))

5.3 Conjunctive Normal Form (7 pts)

Convert all sentences in your knowledge base and the negated sentence that you want to prove to conjunctive normal form. Put **one** clause per line. You may not need to use all lines.

1) (-Tomato(x) v -Ripe(x) v Red(x))

2) (Tomato(x) ^ Ripe(x))

3) (-Tomato(x) v -Red(z))

4)

5)

6)

7)

8)

5.4 Resolution (7 pts)

Using the sentences in conjunctive normal form, use resolution to do proof by contradiction. Do **one** resolution step per line. You may not need to use all lines. Your resolution steps must be of the form: Resolve W with X with unification Y (if unification needs to be done) to obtain Z.

1) Resolve Knowledge Base (KB) 1 with KB 2:

-Tomato(x) v -Ripe(x) v Red(x)
Tomato(x) ^ Ripe(x)

2) Resolve step 1 with KB 3

-Tomato(x) v -Ripe(x) v Red(x)
Tomato(x) ^ Ripe(x)
Tomato(x) → -Red(x)

3) Resolve step 2
-Tomato(x) v -Ripe(x) v Red(x)
Tomato(x) ^ Ripe(x)

Tomato(x) → -Red(x)
Ripe(x) v -Red(x)