Assignments

Course Title: Artificial Intelligence

Course Code: CSE401

Id :16CSE049

Name: Md. Milon Sheikh

Answer to the question no: - 1

[(Food → Party) ∨ (Drinks → Party)] → [(Food ∧ Drinks) → Party]

For the left hand side, we have:

(Food => Party) ∪ (Drinks => Party)

(¬ Food ∪ Party) ∪ (¬ Drinks ∪ Party)

(Food ∪ Party ∪ ¬ Drinks ∪ Party)

(¬Food ¬ Drinks ∪ Party)

And for the right-hand side we have

(Food ∩ Drinks) => Party

(¬Food ∪ ¬Drinks) => Party

(¬Food ∪ ¬ Drinks) Party

(¬Food ∪ ¬ Drinks ∪ Party)

The two sides are identical in CNF, and hence the original is of the form P=>P, which is valid for any P.

Answer to the question no: - 2

Consider a vocabulary with the following symbols:

Occupation (p, o): Predicate. Person p has occupation o.

Customer (p1, p2): Predicate. Person p1 is a customer of person p2.

Boss (p1, p2): Predicate. Person p1 is a boss of person p2.

Doctor, Surgeon, Lawyer, Actor: Constants denoting occupations.

Emily, Joe: Constants denoting people

a)

Assertion: Emily is either a surgeon or a lawyer.

First order logic: *Occupation (Emily, Surgeon)* ∪ *Occupation (Emily, Lawyer)*

b)

Assertion: Joe is an actor, but he also holds another job.

First order logic: $\exists o(o \neq Actor\ ) \cap Occaupation(Joe, Acor) \cap Occupation(Joe, o)$

c)

Assertion: All surgeons are doctors.

First order logic: $\forall p\ Occaupation(p,\ Surgeon) => Occupation(p, Doctor)$

d)

Assertion: Joe does not have a lawyer (i.e., is not a customer of any Lawyer)

First order logic: $\neg\exists p\ Occupation(p,\ Lawyer) \cap Customer(Joe, p)$

e)

Assertion: Emily has a boss who is a lawyer.

First order logic:: $\exists p\ Boss(p,\ Emily) \cap Occupation(p, Lawyer)$

f)

Assertion: There exists a lawyer all of whose customers are doctors.

First order logic:: $\exists p1\ Occupation(p1,\ Lawyer) \cap \forall p2\ Customer(p2, pl) => Occupation(p2, Doctor)$

g)

Assertion: Every surgeon has a lawyer.

First order logic: $\forall p1\ Occupation(p1,\ Surgeon) => \exists p2\ Occupation(p2, Lawyer) \cap Customer\ (p1, p2)$

Answer to the question no: - 3

"Everyones`s DNA is unique and is derived from their parents' DNA."

DNA(x) is the string of DNA characters of person x.

If two people "have the same DNA," it means shared character strings, not shared molecules.

DerivedForm(u,v,w) means strings u is derived from v and w.

$$\forall x, y(\neg (x = y) => \neg (DNA(x) = DNA(y))) \cap DerivedFrom(DNA(x), DNA(Mother(x)), DNA(Father(x)))$$