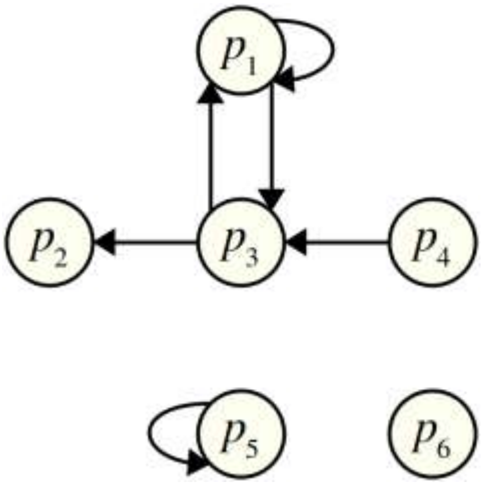
Problem 0:

Prove following statements.

i. p ∨(q ∧ r) is equivalent to (p ∨ q) ∧ (p ∨ r) (Hint: use truth table)
ii. (p→q) ∧ (q→r) is equivalent to (p→r) (Prove using the truth table. And also prove without using the truth table)
iii. Replace ”exclusive OR” operator using the three connectives, ∧ ∨ ¬

Problem 1:

Consider the following diagram:.



If there's an arrow from a person x to a person y, then person x loves person y. We'll denote this by writing Loves(x,y). For example, in this picture, we have Loves(p4,p3) and Loves(p5,p5), but not Loves(p4,p1).

There are no “implied” arrows in this diagram. For example, although p1 loves p3 and p3 loves p2, the statement Loves(p1,p2) is false because there's no arrow from p1 to p2. Similarly, even though p4 loves p3, the statement Loves(p3,p4) is false because there's no arrow from p3 to p4.

Below is a series of first-order logic statements. Some are true, and some are false. Your task is, for each false statement, to tell us the smallest collection of arrows that need to be added to the diagram in order to make the statement true.

We’ve included answers to the first three of these questions as a reference; you need to fill in the rest.

iv. Loves(p1,p3) : True

ii. Loves(p3,p4) : False, add an arrow from P3 to P4

iii. Loves(p1,p2) ∧ Loves(p2,p1) : False add two arrows between P1 and P2

iv. Loves(p1,p2) ∨ Loves(p2,p1)

v. Loves(p1,p1)→Loves(p5,p5)

vi. $\text{Loves}(p1,p2) \rightarrow \text{Loves}(p4,p3)$

vii. $\text{Loves}(p1,p3) \rightarrow \text{Loves}(p3,p6)$

viii. $\text{Loves}(p1,p4) \rightarrow \text{Loves}(p4,p5)$

ix. $\text{Loves}(p1,p4) \leftrightarrow \text{Loves}(p2,p3)$

x. $\text{Loves}(p1,p3) \leftrightarrow \text{Loves}(p5,p5)$

xi. $\forall x. \exists y. \text{Loves}(x,y)$

xii. $\forall x. \exists y. \text{Loves}(y,x)$

xiii. $\forall x. \exists y. (x \neq y \land \text{Loves}(x,y))$

xiv. $\forall x. \exists y. (x \neq y \land \text{Loves}(y,x))$

xv. $\exists x. \forall y. \text{Loves}(x,y)$

xvi. $\exists x. \forall y. (x \neq y \rightarrow \text{Loves}(x,y))$

Problem 2:

In this problem, you'll explore some redundancies within the language of propositional logic.

In lecture, we covered the seven propositional connectives, which for convenience we've listed below:

$\land \ \lor \ \neg \ \rightarrow \ \leftrightarrow \ \top \ \bot$

We settled on this set of connectives because they're convenient and expressive. However, it turns out that four connectives ($\land$, $\lor$, $\neg$, and $\top$.) are enough. That is we can remove following three connectives: $\rightarrow$ $\leftrightarrow$ $\bot$

i. Write expression equivalent to $\bot$ that does not use any connectives besides $\land$, $\lor$, $\neg$, and $\top$. (You're welcome to use parentheses, but do not use any variables.)

ii. Write an expression equivalent to $p \rightarrow q$ that does not use any connectives besides $\land$, $\lor$, $\neg$, and $\top$. (You're welcome to use the variables p and q, along with parentheses.)

iii. Write an expression equivalent to $p \leftrightarrow q$ that does not use any connectives besides $\land$, $\lor$, $\neg$, and $\top$. (You're welcome to use the variables p and q, along with parentheses.)

Your answers to parts (i), (ii), and (iii) of this problem show that the four propositional connectives $\land$, $\lor$, $\neg$, and $\top$ collectively are ***sufficient*** – the other three connectives can be rewritten purely in terms of them. However, there's some redundancy within those four connectives, and we can express all propositional formulas just using three of them.

iv. Write an expression equivalent to $p \lor q$ that does not use any connectives besides $\land$, $\neg$, and $\top$. (You're welcome to use the variables p and q, along with parentheses.)

We can push this further. You can rewrite any propositional formula using just the $\rightarrow$ and $\bot$ connectives!

v. Write an expression equivalent to $\top$ that does not use any connectives besides $\to$ and $\bot$. (You're welcome to use parentheses, but do not use any variables.)

vi. Write an expression equivalent to $\neg p$ that does not use any connectives besides $\to$ and $\bot$. (You're welcome to use the variable p, along with parentheses.)

vii. Write an expression equivalent to $p \land q$ that does not use any connectives besides $\to$ and $\bot$. (You're welcome to use the variables p and q, along with parentheses.)

As a hint, what happens if you negate an implication?

To recap: given the $\to$ and $\bot$ connectives, you can express $\land$, $\neg$, and $\top$. From $\land$, $\neg$, and $\top$ you can get $\land$, $\lor$, $\neg$, and $\top$. And from those four connectives, you can get back the original seven. Overall, any propositional formula can be expressed purely in terms of $\to$ and $\top$.

Problem 3:

For each of the first-order logic formulas below, find a first-order logic formula that is the negation of the original statement. Your final formula must not have any negations in it except for direct negations of predicates. For example, given the formula

$$\forall c.(Cat(c) \to \exists p.(Person(p) \land Loves(p,c))),$$

you could give the formula

$$\exists c.(Cat(c) \land \forall p.(Person(p) \to \neg Loves(p,c))).$$

However, you couldn't give as an answer the formula

$$\exists c.(Cat(c) \land \neg \exists p.(Person(p) \land Loves(p,c))),$$

since the inner negation could be pushed deeper into the expression.

i. Fully negate this formula:

$$\forall p.(Person(p) \to \exists c.(Cat(c) \land Loves(p,c) \land \forall r.(Robot(r) \to \neg Loves(c,r))))$$

ii. Fully negate this formula:

$$(\forall x.(Person(x) \leftrightarrow \exists r.(Robot(r) \land Loves(x,r)))) \to (\forall r.\forall c.(Robot(r) \land Cat(c) \to Loves(r,c)))$$

Be careful – make sure you understand how that formula is parenthesized before you try negating it.

iii. Fully negate this formula:

$$\forall c.(Cat(c) \to \exists r.(Robot(r) \land \forall x.(Loves(c,x) \leftrightarrow r=x)))$$

Problem 4: Translating into Logic (Most important part)

In each of the following, write a statement in first-order logic that expresses the indicated sentence. Your statement may use any first-order construct (equality, connectives, quantifiers, etc.), but you ***must*** only use the predicates Person, Robot, Cat, and Loves (in particular, that means you may *not* use set theory predicates like $\in$ or $\subseteq$!).

i. Write a statement in first-order logic that says "robots do not love."

As a reminder, love is considered directional. Even if robots do not love, it's possible that people or cats might love robots. For example, I could love my Roomba even if it feels nothing toward me.

ii. Write a statement in first-order logic that says "each robot loves every cat, but no cat loves any person."

iii. Write a statement in first-order logic that says "each cat only loves itself."

iv. Write a statement in first-order logic that says "if you pick a person, you'll find that they love a cat if and only if they also love a robot."

v. Write a statement in first-order logic that says "each person loves exactly two cats and nothing else." To clarify, each person is allowed to love a different pair of cats.

vi. Write a statement in first-order logic that says "no two robots love exactly the same set of cats."

Aristotelian logic: Following four classes of statements are called Aristotelian Forms, since they were first described by Aristotle in his work "Prior Analytics"

"All *P*s are *Q*s." $\forall x.\ (P(x) \to Q(x))$	"Some *P*s are *Q*s." $\exists x.\ (P(x) \land Q(x))$
"No *P*s are *Q*s." $\forall x.\ (P(x) \to \neg Q(x))$	"Some *P*s aren't *Q*s." $\exists x.\ (P(x) \land \neg Q(x))$

Assume we are given the following three predicates: Orange(x), Cat(x), Cute(x)

Orange(x): Object x is orange color or not.

Cat(x) and Cute(x) are True if x is a cat and x is cute, respectively.

Now, Translate a sentence: "Eery orange cat is cute".

Which one of the Aristotelian Forms should we use to represent the sentence in an FOL?

Answer: The first one. $\forall$x. (x is an orange cat $\rightarrow$ x is cute)

Let us consider another sentence: "here is a dog that loves everyone."

Assume that we have three predicates; Dog(x), Person(x), Loves(x,y)

Which one should we use? Answer: The second.

There exists x. (x is a dog and x loves everyone) ==

There exists x. (x is a dog and x loves every person y)

There is x. (Dog(x) and (For all y. (y is a Person -> x Loves y)))

Try the next sentence: "Every person loves at least one dog."

Programming and FOL

Consider a Java code that print pairs of numbers in an array.

```
void printPairsIn(vector<int> elems) {
    for (int i = 0; i < elems.size(); i++) {
        for (int j = 0; j < elems.size(); j++) {
            cout << elems[i] << ", " << elems[j] << endl;
        }
    }
}
```

And consider a sentense, “Any two pancakes taste similar.”

Assume that we have two predicates; Pancake(x) and TasteSimilar(x, y).

“Any two pancakes taste similar.”

- ⇨ Any two pancakes x and y taste similar
- ⇨ Any pancake x taste similar to any pancake y
- ⇨ For all x. (Pancake(x) -> x tastes similar to any pancake y)
- ⇨ For all x. (Pancake(x) -> For all y. (Pancake(y) -> x tastes similar to y))

May use a multiple quantifier simultaneously; For all x and y. (x and y are pancakes ->x and y taste similar).

A->B-C is equivalent to (A and B) -> C ?? Tell whether it is True or False. Prove if it is True.

Another example: “Everyone knows at least two people.”

A more complex sentence: “The set of all natural numbers exists.” With predicates Set(x), $x \in y$, Integer(x) and Negative(x).

Optional: Insufficient Connectives

As you saw earlier on this problem set, every propositional logic formula could be written in terms of just $\rightarrow$ and $\perp$. However, you *cannot* express every possible propositional logic formula using just the $\leftrightarrow$ and $\perp$ connectives. Prove why not.