Section 03: Propositions and Proofs

1. Domain Restriction

Translate each of the following sentences into logical notation. These translations require some of our quantifier tricks. You may use the operators $+$ and $\cdot$ which take two numbers as input and evaluate to their sum or product, respectively. Remember:

- To restrict the domain under a $\forall$ quantifier, add a hypothesis to an implication.
- To restrict the domain under an $\exists$ quantifier, AND in the restriction.
- If you want variables to be different, you have to explicitly require them to be not equal.

(a) Domain: Positive integers; Predicates: $\texttt{Even}, \texttt{Prime}, \texttt{Equal}$
"There is only one positive integer that is prime and even."

(b) Domain: Real numbers; Predicates: $\texttt{Even}, \texttt{Prime}, \texttt{Equal}$
"There are two different prime numbers that sum to an even number."

(c) Domain: Real numbers; Predicates: $\texttt{Even}, \texttt{Prime}, \texttt{Equal}$
"The product of two distinct prime numbers is not prime."

(d) Domain: Real numbers; Predicates: $\texttt{Even}, \texttt{Prime}, \texttt{Equal}, \texttt{Postivite}, \texttt{Greater}, \texttt{Integer}$
"For every positive integer, there is a greater even integer"

2. ctrl-z

Translate these logical expressions to English. For each of the translations, assume that domain restriction is being used and take that into account in your English versions.

Let your domain be all UW Students. Predicates $\texttt{143Student}(x)$ and $\texttt{311Student}(x)$ mean the student is in CSE 143 and 311, respectively. $\texttt{BioMajor}(x)$ means x is a bio major, $\texttt{DidHomeworkOne(x)}$ means the student did homwork 1 (of 311). Finally $\texttt{KnowsJava}(x)$ and $\texttt{KnowsDeMorgan}(x)$ mean x knows Java and knows DeMorgan's Laws, respectively.

(a) $\forall x(\texttt{143Student}(x) \rightarrow \texttt{KnowsJava}(x))$

(b) $\exists x(\texttt{143Student}(x) \land \texttt{BioMajor}(x))$

(c) $\forall x([\texttt{311Student}(x) \land \texttt{DidHomeworkOne(x)}] \rightarrow \texttt{KnowsDeMorgan}(x))$

3. Predicate Logic Formal Proof

Given $\forall x.\, T(x) \rightarrow M(x)$, we wish to prove $(\exists x.\, T(x)) \rightarrow (\exists y.\, M(y))$. The following formal proof does this, but it is missing citations for which rules are used, and which lines they are based on. Fill in the blanks with inference rules or predicate logic equivalences, as well as the line numbers.

Then, summarize in English what is going on here.

1. $\forall x. T(x) \to M(x)$ (________________)

2.1. $\exists x. T(x)$	(________________)
Let r be the object that satisfies $T(r)$	
2.2. $T(r)$	(________________, from ________)
2.3. $T(r) \to M(r)$	(________________, from ________)
2.4. $M(r)$	(________________, from ________)
2.5. $\exists y. M(y)$	(________________, from ________)

2. $(\exists x. T(x)) \to (\exists y. M(y))$ (________________, from ________)

4. Quantifier Switch

Consider the following pairs of sentences. For each pair, determine if one implies the other, if they are equivalent, or neither.

(a) $\forall x \ \forall y \ P(x, y)$ $\qquad$ $\forall y \ \forall x \ P(x, y)$

(b) $\exists x \ \exists y \ P(x, y)$ $\qquad$ $\exists y \ \exists x \ P(x, y)$

(c) $\forall x \ \exists y \ P(x, y)$ $\qquad$ $\forall y \ \exists x \ P(x, y)$

(d) $\forall x \ \exists y \ P(x, y)$ $\qquad$ $\exists x \ \forall y \ P(x, y)$

(e) $\forall x \ \exists y \ P(x, y)$ $\qquad$ $\exists y \ \forall x \ P(x, y)$

5. Formal Proof (Direct Proof Rule)

Show that $\neg t \to s$ follows from $t \vee q$, $q \to r$ and $r \to s$.

6. Find the Bug

Each of these inference proofs is incorrect. Identify the line (or lines) which incorrectly apply a law, and explain the error. Then, if the claim is false, give concrete examples of propositions to show it is false. If it is true, write a correct proof.

(a) This proof claims to show that given $a \to (b \vee c)$, we can conclude $a \to c$.

1. $a \to (b \vee c)$ [Given]

2.1. a	[Assumption]
2.2. $\neg b$	[Assumption]
2.3. $b \vee c$	[Modus Ponens, from 1 and 2.1]
2.4. c	[$\vee$ elimination, from 2.2 and 2.3]

2. $a \to c$ [Direct Proof Rule, from 2.1-2.4]

(b) This proof claims to show that given $p \to q$ and r, we can conclude $p \to (q \vee r)$.

$1. p \to q$ [Given]

$2. r$ [Given]

$3. p \to (q \vee r)$ [Intro $\vee$ (1,2)]

(c) This proof claims to show that given $p \to q$ and q that we can conclude p

$1. p \to q$	[Given]
$2. q$	[Given]
$3. \neg p \lor q$	[Law of Implication (1)]
$4. q$	[Eliminate $\lor$ (2,3)

7. Domain Restriction Negated

When we negate a sentence with a domain restriction, the restriction itself remains intact (i.e. not negated) after we have fully simplified. That should make sense; if I claim something is true for every even number, I won't be convinced by you showing me an odd number. In this problem, you'll do the algebra to see why.

(a) Consider the statement $\neg\exists x\exists y(\texttt{Domain1}(x) \land \texttt{Domain2}(y) \land [P(x,y) \land Q(x,y)])$. We know that we should end up with $\forall x\forall y(\texttt{Domain1}(x) \land \texttt{Domain2}(y) \to [\neg P(x,y) \lor \neg Q(x,y)])$ (that is flip the quantifiers, rewrite the domain restriction, and negate the other requirements).

But it can help to see the full algebra written out – write out a step-by-step simplification to get the simplified form (you don't have to label with rules).

(b) Now do the same process for: $\neg\exists x\exists y(\texttt{Domain1}(x) \land \texttt{Domain2}(y) \land [P(x,y) \land Q(x,y)])$

8. Quantifier Ordering

Let your domain of discourse be a set of `Element` objects given in a list called `Domain`. Imagine you have a predicate $\texttt{pred}(x,y)$, which is encoded in the java method `public boolean pred(int x, int y)`. That is you call your predicate `pred` true if and only if the java method returns `true`.

(a) Consider the following Java method:

```
public boolean Mystery(Domain D){
    for(Element x : D) {
        for(Element y : D) {
            if(pred(x,y))
                return true;
        }
    }
}
```

`Mystery` corresponds to a quantified formula (for `D` being the domain of discourse), what is that formula?

(b) What formula does `mystery2` correspond to

```
public boolean Mystery2(Domain D){
    for(Element x : D) {
        boolean thisXPass = false;
        for(Element y : D) {
            if(pred(x,y))
                thisXPass = true;
        }
        if(!thisXPass)
            return false;
    }
    return true;
}
```

9. Formal Proof

Show that $\neg p$ follows from $\neg(\neg r \vee t)$, $\neg q \vee \neg s$ and $(p \to q) \wedge (r \to s)$.

10. A Formal Proof in Predicate Logic

Prove $\exists x\ (P(x) \vee R(x))$ from $\forall x\ (P(x) \vee Q(x))$ and $\forall y\ (\neg Q(y) \vee R(y))$.