

Computer Science 1MD3

Lab 11 – A Quick Introduction To Predicate Logic

Predicate logic is the foundation of mathematics and computer sciences. You have already done some basic logic when dealing with Boolean expressions. The AND and OR operators allowed you to combine conditions to give you greater control, and you have also seen logical definitions of big-oh and big-omega. The purpose of this lab is to give you a formal introduction to predicate logic.

UNARY AND BINARY OPERATORS

Operations like ‘-’ or ‘×’ take element(s) from a set and manipulate them in a certain manner. For instance multiplication takes two numbers x and y and adds x repeatedly to a sum y -many times, and negation, $-$, takes x and negates it so that $x | -x \rightarrow x + (-x) = 0$. Multiplication and negation are examples of a binary and unary operators, respectively. A binary operator, as you may have guessed, operates on two elements from a set ($x | y$) and a unary operator operates on only one ($| x$).

In logic we have the set $\{T, F\}$ that represent true and false. The basic unary operator that we will use is negation, which has the following definition

Negation	Truth Table						
$\neg x \rightarrow \begin{cases} T & \text{if } x = F \\ F & \text{if } x = T \end{cases}$	<table border="1"><thead><tr><th>x</th><th>$\neg x$</th></tr></thead><tbody><tr><td>T</td><td>F</td></tr><tr><td>F</td><td>T</td></tr></tbody></table>	x	$\neg x$	T	F	F	T
x	$\neg x$						
T	F						
F	T						

The basic binary operators are AND and OR.

AND	Truth Table															
$x \wedge y$	<table border="1"><thead><tr><th>x</th><th>y</th><th>$x \wedge y$</th></tr></thead><tbody><tr><td>T</td><td>T</td><td>T</td></tr><tr><td>F</td><td>T</td><td>F</td></tr><tr><td>T</td><td>F</td><td>F</td></tr><tr><td>F</td><td>F</td><td>F</td></tr></tbody></table>	x	y	$x \wedge y$	T	T	T	F	T	F	T	F	F	F	F	F
x	y	$x \wedge y$														
T	T	T														
F	T	F														
T	F	F														
F	F	F														

OR	Truth Table															
$x \vee y$	<table border="1"><thead><tr><th>x</th><th>y</th><th>$x \vee y$</th></tr></thead><tbody><tr><td>T</td><td>T</td><td>T</td></tr><tr><td>F</td><td>T</td><td>T</td></tr><tr><td>T</td><td>F</td><td>T</td></tr><tr><td>F</td><td>F</td><td>F</td></tr></tbody></table>	x	y	$x \vee y$	T	T	T	F	T	T	T	F	T	F	F	F
x	y	$x \vee y$														
T	T	T														
F	T	T														
T	F	T														
F	F	F														

IMPLICATION AND BI-IMPLICATION

Implication and bi-implication are binary operators that we haven’t seen before. We will first deal with implication, which is represented by \rightarrow . $x \rightarrow y$ reads x implies y , and may be understood in any of the following ways:

If p , then q
 p is sufficient for q
 q if p
 q when p

p implies q
 p only if q
 q whenever p
 q is necessary for p

If p , q
 P is sufficient for q
 q when p
 q follows from p

IMPLIES	Truth Table															
$x \rightarrow y$	<table border="1"> <thead> <tr> <th>x</th> <th>y</th> <th>$x \rightarrow y$</th> </tr> </thead> <tbody> <tr> <td>T</td> <td>T</td> <td>T</td> </tr> <tr> <td>F</td> <td>T</td> <td>T</td> </tr> <tr> <td>T</td> <td>F</td> <td>F</td> </tr> <tr> <td>F</td> <td>F</td> <td>T</td> </tr> </tbody> </table>	x	y	$x \rightarrow y$	T	T	T	F	T	T	T	F	F	F	F	T
x	y	$x \rightarrow y$														
T	T	T														
F	T	T														
T	F	F														
F	F	T														

Bi-implication, represented by \leftrightarrow , where $x \leftrightarrow y$ reads x bi-implies y. Bi-implication may be understood in the following ways:

p is necessary and sufficient for q *If p then q, and conversely* *p if q*

BI-IMPLIES	Truth Table															
$x \leftrightarrow y$	<table border="1"> <thead> <tr> <th>x</th> <th>y</th> <th>$x \leftrightarrow y$</th> </tr> </thead> <tbody> <tr> <td>T</td> <td>T</td> <td>T</td> </tr> <tr> <td>F</td> <td>T</td> <td>F</td> </tr> <tr> <td>T</td> <td>F</td> <td>F</td> </tr> <tr> <td>F</td> <td>F</td> <td>T</td> </tr> </tbody> </table>	x	y	$x \leftrightarrow y$	T	T	T	F	T	F	T	F	F	F	F	T
x	y	$x \leftrightarrow y$														
T	T	T														
F	T	F														
T	F	F														
F	F	T														

LOGICAL EQUIVLANCES

$x \wedge y$, $x \rightarrow y$, $\neg x$ are all examples of unary or binary operations, when one or two symbols are being operated on, we call the expression a *proposition*. We say that the propositions p, and q are logically equivalent if $p \leftrightarrow q$ is a tautology, or in other words, it will always evaluate to true. The notation $p \equiv q$ denotes that p and q are logically equivalent.

One way to determine whether two propositions are equivalent is to use a truth table. In particular, the propositions p and q are equivalent if and only if the columns giving their truth-value agree. The following example illustrates this method.

Example 1

Show that $\neg(p \vee q)$ and $\neg p \wedge \neg q$ are logically equivalent.

p	q	$\neg(p \vee q)$	$\neg p \wedge \neg q$	$\neg(p \vee q) \leftrightarrow \neg p \wedge \neg q$
T	T	$\neg(T) \equiv F$	$\neg T \wedge \neg T = F \wedge F$ $= F$	$F \leftrightarrow F \equiv T$
T	F	$\neg(T) \equiv F$	$\neg T \wedge \neg F = F \wedge T$ $= F$	$F \leftrightarrow F \equiv T$
F	T	$\neg(T) \equiv F$	$\neg F \wedge \neg T = T \wedge F$ $= F$	$F \leftrightarrow F \equiv T$
F	F	$\neg(F) \equiv T$	$\neg F \wedge \neg F = T \wedge T$ $= T$	$T \leftrightarrow T \equiv T$

Since $\neg(p \vee q) \leftrightarrow \neg p \wedge \neg q$ always evaluates to True we say these propositions are logically equivalent.

QUANTIFIERS

It is possible to have binary statements that are true only sometimes, consider $P(x) : x > 6$, this must evaluate to either true or false. In order to evaluate this statement, we employ the quantifiers \forall, \exists , which respectively read *for all* and *there exists*.

So what does $\forall x P(x) = \forall x (x > 6)$ evaluate too? In plain English it reads: for all x , x is greater than 6, so it clearly evaluates to false since 5 isn't greater than 6. How about $\exists x (x > 6)$, which reads: there exists an x where $x > 6$. Well if $x=7$, 7 is greater than 6, so this must be true.

<i>Statement</i>	<i>When True</i>	<i>When False</i>
$\forall x P(x)$	$P(x)$ is true for every x	There is an x for which $P(x)$ is false
$\exists x P(x)$	There is an x for which $P(x)$ is true	$P(x)$ is false for every x

It is also possible to negate these quantifiers. Just think of it as whenever you move a negation "past" a quantifier that you must switch it from *there exists* to a *for all* and vice versa.

<i>Negation</i>	<i>Equivalent Statement</i>	<i>When Is Negation True</i>	<i>When False</i>
$\neg \exists P(x)$	$\forall x \neg P(x)$	For every x , $P(x)$ is false.	There is an x for which $P(x)$ is true.
$\neg \forall x P(x)$	$\exists x \neg P(x)$	There is an x for which $P(x)$ is false.	$P(x)$ is true for every x .

It is also possible to use more than one quantifier on several variables. This is called nesting the quantifier.

<i>Statement</i>	<i>When True</i>	<i>When False</i>
$\forall x \forall y P(x, y)$ $\forall y \forall x P(x, y)$	$P(x, y)$ is true for every pair x, y	There is a pair x, y for which $P(x, y)$ is false.
$\forall x \exists y P(x, y)$	For every x there is a y for which $P(x, y)$ is true	There is an x such that $P(x, y)$ is false for every y .
$\exists x \forall y P(x, y)$	There is an x for which $P(x, y)$ is true for every y .	For every x there is a y for which $P(x, y)$ is false.
$\exists x \exists y P(x, y)$ $\exists y \exists x P(x, y)$	There is a pair x, y for which $P(x, y)$ is true.	$P(x, y)$ is false for every pair x, y .

Examples.

For the proposition $P(x, y) : x > y$ where $x, y \in \mathbb{Z}$ what do the following expression evaluate too?

$\forall x \forall y P(x, y) = F$ since for any x I can choose a y where this fails.

$\neg \forall x \exists y P(x, y) \equiv \exists x \neg \exists y P(x, y) \equiv \exists x \forall y \neg P(x, y) = F$ since there does not exist x for all y such that $P(x, y)$ always fails.

$\forall y \exists x P(x, y) = T$ since for any selection of y I can select x where $P(x, y)$ evaluates to true.

Self test problems:

Let p and q be the propositions

p : It is below freezing

q : It is snowing

Write the propositions using p and q and logical connectives.

- It is below freezing and snowing.
 - It is below freezing but not snowing.
 - It is not below freezing and it is not snowing.
 - It is either not snowing or below freezing (or both)
 - It is below freezing or it is snowing, but it is not snowing if it is below freezing.
 - That it is below freezing is necessary and sufficient for it not to be snowing.
-

Construct a truth table for each of the these compound propositions.

- $p \rightarrow (\neg q \vee r)$
 - $\neg p \rightarrow (q \rightarrow r)$
 - $(p \rightarrow q) \wedge (\neg p \rightarrow r)$
-

Use a truth table to verify logical equivalence

$$\neg(p \wedge q) \equiv \neg p \vee \neg q$$

Use a truth table to show that the following are tautologies (always evaluate to true).

- $(p \wedge q) \rightarrow p$
 - $p \rightarrow (p \vee q)$
 - $\neg p \rightarrow (p \rightarrow q)$
-

Determine the truth of the following, assuming that in all instances $n \in \mathbb{Z}$.

- $\forall n(n = 1 > n)$
 - $\exists n(2n = 3n)$
 - $\exists n(n = -n)$
 - $\forall n(n^2 \geq n)$
-

Translate these statements into English, where the universe of discourse for each variable consists of all real numbers.

- $\exists x \forall y (xy = y)$
- $\forall x \forall y (((x \geq 0) \wedge (y < 0)) \rightarrow (x - y > 0))$
- $\forall x \forall y \exists z (x = y + x)$