CSPC15 PRELIMS

Propositional Logic

- Construct correct mathematical arguments

Proposition – a declarative sentence that is either true or false, but not true and false.

Uses:
- Designing Circuits
- Creating Programs
- Validating Programs

Examples:
- The sky is blue.
- The moon is made of cheese.
- Luke, I am your father.
- Sit Down.
- x + 1 = 2

Compound Propositions

- A compound proposition is comprised of propositions and one or more of the following connectives:

Negation	¬	NOT
Conjunction	∧	AND
Disjunction	∨	OR
Implication	→	IF THE
Biconditional	↔	IF AND ONLY IF

- Typically use letters: p, q, r, s, t, etc.
- True = T (also "1")
- False = F (also "0")

Negation – equals to the opposite truth value of the proposition.

- ¬ , ─
- eg., ¬p , p
- pronounce: "not p"

Example:

Given:
1. A.R. was in *Sharknado 4: The 4th Awakens*
2. Nicholas Cage was in *Matchstick Men*

Negation:
1. A.R. was NOT in *Sharknado 4: The 4th Awakens*
2. Nicholas Cage was NOT in *Matchstick Men*

Or:
1. It is NOT the case that A.R. was in *Sharknado 4: The 4th Awakens*
2. It is NOT the case that Nicholas Case was in *Matchstick Men*

Truth Table

- Each row of a truth table give us one possibility for the truth values of our proposition(s). Since each proposition has two possible truth values, true or false, we will have 2 rows for each propositions (or 2n rows where n is the number of propositions)

p	¬p
T	F
F	T

Conjunction

- The conjunction of propositions p and q is denoted p∧q and read p "AND" q. For conjunction to be true, both propositions must be true.

p	q	p∧q
T	T	T
T	F	F
F	T	F
F	F	F

Disjunction

- The disjunction of propositions p and q is denoted p∨q and read p "OR" q. For disjunction to be true, either proposition must be true.

p	q	p∨q
T	T	T
T	F	T
F	T	T
F	F	F

The connective "OR" in English "XOR"

- **Inclusive "OR" p ∨ q**

  The prerequisite for MA420 is either Ma315 or MA335.

- **Exclusive "OR" p ⊕ q**

  You get soup or salad with your entrée.

p	q	p∨q	p ⊕ q
T	T	T	F
T	F	T	T
F	T	T	T
F	F	F	F

Implications

- Has a "FALSE" truth value when the first proposition is "TRUE" and the second proposition is "FALSE". Otherwise, it has a "TRUE" truth value.

p	q	p→q
T	T	T
T	F	F
F	T	T
F	F	T

Biconditional
- Has a "TRUE" truth value when both propositions share same truth value. Otherwise, it has a "FALSE" truth value.

p	q	p ↔ *q*
T	T	T
T	F	F
F	T	F
F	F	T

Rows
- Need a row for every possible combination of values for the compound proportions.

Columns
- Need a column for each propositional variable.
- Need a column for the truth value of each expression that occurs in the compound proposition as it is built up.
- Need a column for the compound proposition (usually at far right).

Order of Operators

PRECEDENCE	OPERATOR
1	¬
2	∧
3	∨
4	→
5	↔

Connectives	Word Form	Statement	Symbolic Form
Negation	not	not p	¬p
Conjunction	and	p and q	p ∧ q
Disjunction	or	p or q	p ∨ q
Implication	If…then	If p…then q	p → q
Biconditional	If and only if	p if and only if q	p ↔ q

Logical Expression

- Any proposition must somehow be expressed verbally, graphically, or by a string of characters.
- A proposition expressed by a string of characters is called a **logical expression** or a **formula**.
- It can either be atomic or compound.
  - **Atomic expression** is consisting of a single propositional variable and it represents atomic proposition.
  - **Compound expressions** contains at least **one connective**, and they represent compound proportions.
- All expressions containing identifiers that represent expressions are called **schemas.**
- If A=(P∧Q) and if B=(P∨Q), then the schema (A→B) stands for ((P∧Q)→(P∨Q)).

Precedence Rules

1. ~
2. ^
3. ∨
4. →
5. ↔

- ~P ∨ Q is equal to (~P) ∨ Q not ~(P ∨ Q)
- P ^ Q ∨ R is equal to (P ^ Q) ∨ R
- P → Q ∨ R is equal to P → (Q v R)
- M ↔ F → C is equal to M ↔ (F → C)
- P → Q → R is equal to (P → Q) → R

Because all binary logical connectives are **left associative**.

Prefix
- precedes it operand (start)

Infix
- inserted between the operands (middle)

Postfix
- follows its operands (end)

Evaluation of Expressions and Truth Tables

Prefix Expression	**Infix Expression**	**Postfix Expression**
+ a b, ~ p	a + b, p ^ q	a b +
+ a * b c, ~ q	a + b * c, p ^ q ∨ r	a b c * +

- "If you take a class in computer, and if you do not understand logic, you will not pass."

P: You take a class in computer.
Q: You understand logic.
R: You pass.

- We want to know exactly when this statement is true or false. To do this we use the symbols P, Q, and R. Using this, the statement in question becomes **(P ^ ~Q) → ~R**.

Truth Table for (P ^ ~Q) → ~R

P	**Q**	**R**	**~Q**	**P^~Q**	**~R**	**(P ^ ~Q) → ~R**
T	T	T	F	F	F	T
T	T	F	F	F	T	T
T	F	T	T	T	F	F
T	F	F	T	T	T	T
F	T	T	F	F	F	T
F	T	F	F	F	T	T
F	F	T	T	F	F	T
F	F	F	T	F	T	T

Tautologies and Contradictions

Tautology
- gives all answers with truth value of TRUE.

Contradiction
- gives all answers with truth value of FALSE.

Contingency
- proposition that is neither a tautology nor a contradiction.

Tautology and Logical equivalence Definitions:
- A compound proposition that is always True is called a **tautology**.
- Two propositions p and q are logically equivalent if their truth tables are **the same**.
- Namely, p and q are logically equivalent if p ↔ q is a tautology. If p and q are logically equivalent, we write **p ≡ q.**

Examples of Logical Equivalence

Example:
Look at the following two compound propositions:
p → q and **q ∨ ¬p**.

p	q	p → q
T	T	T
T	F	F
F	T	T
F	F	T

p	q	¬p	q ∨ ¬p
T	T	F	T
T	F	F	F
F	T	T	T
F	F	T	T

- The last column of the two truth tables are identical. Therefore (p → q) and (q ∨ ¬p) are logically equivalent.
- So (p → q) ↔ (q ∨ ¬p) is a **tautology**.
- Thus: **(p → q) ≡ (q ∨ ¬p)**.

De Morgan Law

- We have a number of rules for logical equivalence.
- For example:
¬(p ^ q) ≡ ¬p ∨ ¬q

p	q	p ^ q	¬(p ^ q)
T	T	T	F
T	F	F	T
F	T	F	T
F	F	F	T

p	q	¬p ∨ ¬q
T	T	F
T	F	T
F	T	T
F	F	T

Distributivity
p ∨ (q ∧ r) ≡ (p ∨ q) ∧ (p ∨ r)

p	q	r	q ∧ r	p ∨ (q ∧ r)
T	T	T	T	T
T	T	F	F	T
T	F	T	F	T
T	F	F	F	T
F	T	T	T	T
F	T	F	F	F
F	F	T	F	F
F	F	F	F	F

p	q	r	p ∨ q	p ∨ r	p ∨ (q ∧ r)
T	T	T	T	T	T
T	T	F	T	T	T
T	F	T	T	T	T
T	F	F	T	T	T
F	T	T	T	T	T
F	T	F	T	F	F
F	F	T	F	T	F
F	F	F	F	F	F

Contrapositves
- The proposition **¬q → ¬p** is called the **Contrapositive** of the proposition **p → q**. They are logically equivalent.
p → q ≡ ¬q → ¬p

p	q	p → q
T	T	T
T	F	F
F	T	T
F	F	T

p	q	¬q → ¬p
T	T	T
T	F	F
F	T	T
F	F	T

Logic Equivalences

Equivalence	Name
p ^ T ≡ p, p v F ≡ p	Identity laws
p v T ≡ T, p ^ F ≡ F	Domination laws
p v p ≡ p, p ^ p ≡ p	Idempotent laws
¬ (¬p) ≡ p	Double negation law
p v q ≡ q v p p ^ q ≡ q ^ p	Commutative laws
(p v q) v r ≡ p v (q v r) (p ^ q) ^ r ≡ p ^ (q ^ r)	Associative laws
p v (q ^ r) ≡ (p v q) ^ (p v r) p ^ (q v r) ≡ (p ^ q) v (p ^ r)	Distributive laws
¬ (p v q) ≡ ¬p ^ ¬q ¬ (p ^ q) ≡ ¬p v ¬q	De Morgan's laws
p v (p ^ q ≡ p p ^ (p v q) ≡ p	Absorption laws
p v ¬p ≡ T, p ^ ¬p ≡ F	Negation laws

Predicate Logic

Predicate Logic
- deals with predicates, which are propositions containing variables.

Predicate
- is an expression of one or more variables defined on some specific domain. A predicate with variables can be made a proposition by either assigning a value to the variable or by quantifying the variable.
- is a property that is affirmed or denied about the subject (in logic, we say "variable" or "argument") of a statement.
- is a property that is affirmed or denied about the subject (in logic, we say "variable" or "argument") of a statement.

Example:
- Let E(x, y) denote "x = y"
- Let X(a, b, c) denote "a + b + c = 0"
- Let M(x, y) denote "x is married to y"

$\underbrace{x}_{\text{Subject}}\ \underbrace{\textit{is greater than 3}}_{\text{Predicate}}$

We introduce a (functional) symbol for the predicate, and put the subject as an argument (to the functional symbol): P(x)

Example:

- Father(x): unary predicate
- Brother(x,y): binary predicate
- Sum(x,y,z): ternary predicate
- P(x,y,z,t): n-ary predicate

A statement of the form P(x1, x2, . . . , xn) is the value of the propositional function P. Here, (x1, x2, . . . , xn) is an n-tuple and P is a predicate.

You can think of a propositional function as a function that

- Evaluates to true or false.
- Takes one or more arguments.
- Expresses a predicate involving the argument(s).
- Becomes a proposition when values are assigned to the arguments.

Example:

Let Q(x, y, z) denote the statement "x2 + y2 = z2". What is the truth value of Q(3, 4, 5)? What is the truth value of Q(2, 2, 3)? How many values of (x, y, z) make the predicate true?

Since 9 + 16 = 25 → **25= 25** Q(3, 4, 5) is true.

Since 4 + 4 = 9, → **8 = 9** Q(2, 2, 3) is false.

There are infinitely many values for (x, y, z) that make this propositional function true—how many right triangles are there?

Quantifiers

- A predicate becomes a proposition when we assign it with fixed values. However, another way to make a predicate into a proposition is to quantify it. That is, the predicate is true (or false) for all possible values in the universe of discourse or for some value(s) in the universe of discourse.

Universal Quantifiers

- The universal quantification of a predicate P(x) is the proposition "P(x) is true for all values of x in the universe of discourse" We use the notation

∀xP(x) - which can be read "for all x"

- If the universe of discourse is finite, say {n1, n2, . . . , nk}, then the universal quantifier is simply the conjunction of all elements:

∀ xP(x) () $P(n_1)$ ^ P(n2) ^ · · · ^ $P(n_k)$

Example

"Man is mortal" can be transformed into the propositional form ∀xP(x) where P(x) is the predicate which denotes x is mortal and the universe of discourse is all men.

Existential Quantifiers

- Existential quantifier states that the statements within its scope are true
for some values of the specific variable. It is denoted by the symbol ∃

∃xP(x) is read as for some values of x, P(x) is true.

Example

"Some people are dishonest" can be transformed into the propositional form ∃xP(x) where P(x) is the predicate which denotes x is dishonest and the universe of discourse is some people.

- The existential quantification of a predicate P(x) is the proposition "There exists an x in the universe of discourse such that P(x) is true."We use the notation

∃xP(x) which can be read "there exists an x"

- Again, if the universe of discourse is finite, {n1, n2, . . . , nk}, then the existential quantifier is simply the disjunction of all elements:

∃ xP(x) () P(n1) _ P(n2) _ · · · _ P(nk)

Quantifiers: Truth Values

In general, when are quantified statements true/false?

Statement	True When	False When
∀xP(x)	P(x) is true for every x.	There is an x for which P(x) is false.
∃xP(x)	There is an x for which P(x) is true.	P(x) is false for every x.

Mixing Quantifiers

- Existential and universal quantifiers can be used together to quantify a predicate statement; for example,

∀x∃yP(x, y)

is perfectly valid. However, you must be careful—it must be read left to
right.
For example, ∀x∃y P(x, y) is not equivalent to ∃x∀y P(x, y). Thus, ordering is important.

Example:

- ∀x ∃y Loves(x, y) : everybody loves somebody
- ∃x∀y Loves(x, y) : There is someone loved by everyone

Mixing Quantifiers: Truth Values

Statement	True When	False When
∀x∀y P(x, y)	P(x, y) is true for every pair x, y.	There is at least one pair, x, y for which P(x, y) is false.
∀x∃y P(x, y)	For every x, there is a y for which P(x, y) is true.	There is an x for which P(x, y) is false for every y.
∃x∀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.
∃x∃y P(x, y)	There is at least one pair x, y for which P(x, y) is true.	P(x, y) is false for every pair x, y.

Binding Variables

- When a quantifier is used on a variable x, we say that x is bound. If no quantifier is used on a variable in a predicate statement, it is called free.

Example:

- In the expression $\exists x \forall yP(x, y)$ both x and y are bound.
- In the expression $\forall xP(x, y)$, x is bound, but y is free.

- A statement is called a well-formed formula, when all variables are properly quantified.

- The set of all variables bound by a common quantifier is the scope of that quantifier.

Example:

In the expression $\exists x,y \forall zP(x, y, z, c)$ the scope of the existential quantifier is $\{x, y\}$, the scope of the universal quantifier is just z and c has no scope since it is free.

Good luck, CS 1-2!