Discrete Structures

Topic 4 – Logic: Predicate Logic

(Ch. 1.4)*

CMPS 211 – American University of Beirut

* Extracted from *Discrete Mathematics and It's Applications* book slides

Propositional Logic Not Enough

- If we have
  - “All men are mortal”
  - “Socrates is a man”
- Does it follow that “Socrates is mortal?”
  - Well, logically it does!
- However, it can’t be deducted using propositional logic
  - Need a language that talks about objects, their properties, and their relations and allow us to draw inferences

Introducing Predicate Logic

- Predicate logic (or first-order logic in general) is a formal system for logical reasoning about objects, by using the following new features:
  - Variables: x, y, z
  - Predicates: P, M, R
  - Propositional functions: P(x), M(x, y)
  - Quantifiers: ∀, ∃

Propositional Functions

- Propositional functions are a generalization of propositions
  - They contain a predicate and variables
  - Each variable has a domain and can be replaced by elements from its domain
  - Propositional functions become propositions (and have truth values) when their variables are each replaced by a value from the domain
- For example, let P(x) denote "$x > 0$", where x is the variable, "greater than zero" is the predicate and the domain is the set of integers, then
  - P(-3) is false
  - P(0) is false
  - P(3) is true

Propositional Functions Exercise

- Let “$x + y = z$” be denoted by $R(x, y, z)$ and the domain U (for all three variables) be the set of integers

- Find the truth values for
  - R(2,-1,5)
  - R(3,4,7)
  - R(x, 3, z)

Compound Expressions

- Connectives from propositional logic carry over to predicate logic
- If $P(x)$ denotes "$x > 0$", then
  - $P(3) \vee P(-1)$ is true
  - $P(3) \wedge P(-1)$ is false
  - $P(3) \rightarrow P(-1)$ is false
  - $P(3) \rightarrow P(1)$ is true
- Expressions with variables are not propositions and therefore do not have truth values
- For example:
  - $P(3) \wedge P(y)$
  - $P(x) \rightarrow P(y)$

Quantifiers

- We need quantifiers to express the meaning of English words including all and some
  - "All men are Mortal"
  - "Some cats do not have fur"

Charles Peirce
(1839-1914)

- The two most important quantifiers are
  - Universal Quantifier "For all" with symbol: $\forall$
  - Existential Quantifier "There exists" with symbol: $\exists$
  - There are several other quantifiers like exactly 1, 2 or more, and so on (but we won't cover/use in this course)
- The quantifiers are said to bind the variable x in these expressions

Universal Quantifier

- $\forall x\, P(x)$ is read as "For all x, P(x)" or "For every x, P(x)"
- $\forall x\, P(x)$ asserts P(x) is true for every x in the domain

- Examples:
  - If P(x) denotes "$x > 0$" and U is the set of integers,
then $\forall x\, P(x)$ is false

  - If P(x) denotes "$x > 0$" and U is the set of positive integers,
then $\forall x\, P(x)$ is true

  - If P(x) denotes "x is even" and U is the set of integers
then $\forall x\, P(x)$ is false

Existential Quantifier

- $\exists x\, P(x)$ is read as "For some x, P(x)", or as "There is an x such that P(x)," or "For at least one x, P(x)"
- $\exists x\, P(x)$ asserts P(x) is true for some x in the domain

- Examples:
  - If P(x) denotes "$x > 0$" and U is the set of integers, then $\exists x\, P(x)$ is true
  - If P(x) denotes "$x < 0$" and U is the set of positive integers, then $\exists x\, P(x)$ is false
  - If P(x) denotes "x is even" and U is the set of integers, then $\exists x\, P(x)$ is true

Properties of Quantifiers

- The truth value of $\exists x\, P(x)$ and $\forall x\, P(x)$ depend on both the propositional function $P(x)$ and on the domain U

- Examples:
  - If U is the set of positive integers and $P(x)$ is the statement "$x < 2$", then $\exists x\, P(x)$ is true, but $\forall x\, P(x)$ is false
  - If U is the set of negative integers and $P(x)$ is the statement "$x < 2$", then both $\exists x\, P(x)$ and $\forall x\, P(x)$ are true
  - If U consists of 3, 4, and 5, and $P(x)$ is the statement "$x < 2$", then both $\exists x\, P(x)$ and $\forall x\, P(x)$ are false

    But if $P(x)$ is the statement "$x > 2$", then both $\exists x\, P(x)$ and $\forall x\, P(x)$ are true

Precedence of Quantifiers

- The quantifiers ∀ and ∃ have higher precedence than all the logical operators
  - For example, $\forall x\ P(x) \lor Q(x)$ means $(\forall x\ P(x)) \lor Q(x)$
  - $\forall x\ (P(x) \lor Q(x))$ means something different

- It is a common mistake to write $\forall x\ P(x) \lor Q(x)$ when you mean $\forall x\ (P(x) \lor Q(x))$

Translating from English to Logic

- Example 1:
  - Translate the following sentence into predicate logic: "Every student in this class has taken a course in Java"

- Solution:
  - First decide on the domain U
  - Solution 1: If U is all students in this class, define a propositional function J(x) denoting "x has taken a course in Java" and translate as $\forall x\, J(x)$
  - Solution 2: But if U is all people, also define a propositional function S(x) denoting "x is a student in this class" and translate as $\forall x\, (S(x) \rightarrow J(x))$

$\forall x\, (S(x) \wedge J(x))$ is not correct. What does it mean?

Translating from English to Logic

- Example 2:
  - Translate the following sentence into predicate logic: "Some student in this class has taken a course in Java"
- Solution:
  - First decide on the domain U
  - Solution 1: If U is all students in this class, translate as $\exists x\, J(x)$
  - Solution 2: But if U is all people, then translate as $\exists x\, (S(x) \land J(x))$

$\exists x\, (S(x) \rightarrow J(x))$ is not correct. What does it mean?

Translating from English to Logic

More Examples:

- "Some student in this class has visited Mexico"
- Solution: Let M(x) denote "x has visited Mexico" and S(x) denote "x is a student in this class," and U be all people

$$\exists x\ (S(x) \land M(x))$$

- "Every student in this class has visited Canada or Mexico"
- Solution: Add C(x) denoting "x has visited Canada"

$$\forall x\ (S(x) \rightarrow (M(x) \lor C(x)))$$

Equivalences in Predicate Logic

- Statements involving predicates and quantifiers are logically equivalent if and only if they have the same truth value
  - for every predicate substituted into these statements and
  - for every domain of discourse used for the variables in the expressions
- The notation $S \equiv T$ indicates that S and T are logically equivalent
- Example:
  - $\forall x\, (P(x) \wedge Q(x)) \equiv \forall x\, P(x) \wedge \forall x\, Q(x)$

Thinking about Quantifiers as Conjunctions and Disjunctions

- If the domain is finite,
  - a universally quantified proposition is equivalent to a conjunction of propositions without quantifiers
  - and an existentially quantified proposition is equivalent to a disjunction of propositions without quantifiers

- For example, if U consists of the integers 1, 2 and 3:

$$\forall x P(x) \equiv P(1) \wedge P(2) \wedge P(3)$$

$$\exists x P(x) \equiv P(1) \vee P(2) \vee P(3)$$

- Even if the domains are infinite, you can still think of the quantifiers in this fashion, but the equivalent expressions without quantifiers will be infinitely long

Negating Quantified Expressions

- Consider $\forall x\, J(x)$
  - "Every student in your class has taken a course in Java"
  - Here J(x) is "x has taken a course in Java" and

  the domain is students in your class

  - Negating the original statement gives "It is not the case that every student in your class has taken Java"
  - This implies that "There is a student in your class who has not taken Java"

- Symbolically $\neg\forall x\, J(x)$ and $\exists x\, \neg J(x)$ are equivalent

Negating Quantified Expressions (cont.)

- Now Consider $\exists x J(x)$
  - "There is a student in this class who has taken a course in Java"
  - Here $J(x)$ is "x has taken a course in Java" and

    the domain is students in your class

  - Negating the original statement gives "It is not the case that there is a student in this class who has taken Java"
  - This implies that "Every student in this class has not taken Java"
- Symbolically $\neg\exists x J(x)$ and $\forall x \neg J(x)$ are equivalent

De Morgan's Laws for Quantifiers

- The rules for negating quantifiers are

TABLE 2 De Morgan's Laws for Quantifiers.			
Negation	***Equivalent Statement***	***When Is Negation True?***	***When False?***
$\neg\exists x\, P(x)$	$\forall x \neg P(x)$	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.

- The reasoning in the table shows that

$$\neg\forall x P(x) \equiv \exists x \neg P(x)$$

$$\neg\exists x P(x) \equiv \forall x \neg P(x)$$

Translation Exercise

- U = {fleegles, snurds, thingamabobs}
  - F(x): x is a fleegle
  - S(x): x is a snurd
  - T(x): x is a thingamabob

- Translate "Everything is a fleegle"

- Solution: $\forall x \, F(x)$

Translation Exercise (cont.)

- U = {fleegles, snurds, thingamabobs}
  - F(x): x is a fleegle
  - S(x): x is a snurd
  - T(x): x is a thingamabob

- Translate "Nothing is a snurd"
- Solution: $\neg \exists x\, S(x)$

- What is this equivalent to?
- Solution: $\forall x \neg S(x)$

Translation Exercise (cont.)

- U = {fleegles, snurds, thingamabobs}
  - F(x): x is a fleegle
  - S(x): x is a snurd
  - T(x): x is a thingamabob

- Translate "All fleegles are snurds"

- Solution: $\forall x\ (F(x) \rightarrow S(x))$

Translation Exercise (cont.)

- U = {fleegles, snurds, thingamabobs}
  - F(x): x is a fleegle
  - S(x): x is a snurd
  - T(x): x is a thingamabob

- Translate "Some fleegles are thingamabobs"

- Solution: $\exists x\ (F(x) \land T(x))$

Translation Exercise (cont.)

- U = {fleegles, snurds, thingamabobs}
  - F(x): x is a fleegle
  - S(x): x is a snurd
  - T(x): x is a thingamabob

- Translate "No snurd is a thingamabob"
- Solution: $\neg\exists x\ (S(x) \land T(x))$

- What is this equivalent to?
- Solution: $\forall x\ (\neg S(x) \lor \neg T(x))$

Translation Exercise (cont.)

- U = {fleegles, snurds, thingamabobs}
  - F(x): x is a fleegle
  - S(x): x is a snurd
  - T(x): x is a thingamabob

- Translate "If any fleegle is a snurd then it is also a thingamabob"

- Solution: $\forall x\ ((F(x) \land S(x)) \rightarrow T(x))$

System Specification Example

- Predicate logic is used for specifying properties that systems must satisfy
- For example, translate into predicate logic
  - "Every mail message larger than one megabyte will be compressed"
  - "If a user is active, at least one network link will be available"
- Solution
  - Let L(m, y) be "Mail message m is larger than y megabytes"
  - Let C(m) denote "Mail message m will be compressed"
  - Let A(u) represent "User u is active"
  - Let S(n, x) represent "Network link n is state x"
- Now we have:

$$\forall m(L(m,1) \to C(m))$$
$$\exists u\, A(u) \to \exists n\, S(n, available)$$

Any Questions?