CS 2305: Discrete Mathematics for Computing I

Lecture 09

- KP Bhat

Order of Quantifiers (4)

Example 3: Let *U* be the real numbers,

Define $P(x,y) : x \cdot y = 0$

What is the truth value of the following:

1. $\forall x \forall y P(x, y)$

   Answer:False

2. $\forall x \exists y P(x, y)$

   Answer:True

3. $\exists x \forall y P(x, y)$

   Answer:True

4. $\exists x \exists y P(x, y)$

   Answer:True

Order of Quantifiers (5)

Example 4: Let *U* be the real numbers,

Define *P(x,y)* : *x* / *y* = 1

What is the truth value of the following:

1. $\forall x \forall y P(x, y)$

   Answer:False

2. $\forall x \exists y P(x, y)$

   Answer:False

3. $\exists x \forall y P(x, y)$

   Answer:False

4. $\exists x \exists y P(x, y)$

   Answer:True

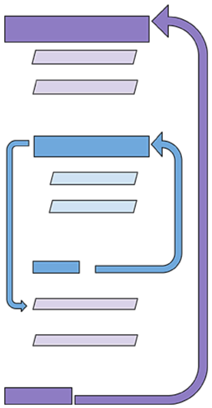
Quantifications of Two Variables

TABLE 1 Quantifications of Two Variables.

Statement	***When True?***	***When False?***
$\forall x \forall y P(x, y)$ $\forall y \forall x P(x, y)$	$P(x, y)$ 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.

Two Surprising Results (1)

- If $\exists y \forall x P(x, y)$ is true, then $\forall x \exists y P(x, y)$ must also be true
- If $\forall x \exists y P(x, y)$ is true, it is not necessary for $\exists y \forall x P(x, y)$ to be true
- We will demonstrate (not prove) these results using the "nested loops" abstraction



Two Surprising Results (2)

If $\exists y \forall x P(x, y)$ is true, then $\forall x \exists y P(x, y)$ must also be true

Let us assume that the 5th iteration of the outer loop is a "magical" iteration in that for all values of the inner loop, $P(x,y)$ evaluates to true.

If we now switch the order of the loops (i.e. the previously outer loop becomes the inner loop and vice-versa) we know that for all iterations of the outer loop, the 5th iteration of the inner loop evaluates to true.

Hence $\forall x \exists y P(x, y)$ is true

Two Surprising Results (3)

If $\forall x \exists y P(x, y)$ is true, it is not necessary for $\exists y \forall x P(x, y)$ to be true

Let us assume that for each iteration of the outer loop, a different iteration of the inner loop evaluates $P(x,y)$ to true.

If we now switch the order of the loops (i.e. the previously outer loop becomes the inner loop and vice-versa) we know that there is no iteration of the outer loop for which each iteration of the inner loop evaluates $P(x,y)$ to true.

Hence we cannot make the claim that $\exists y \forall x P(x, y)$ is true

Translating Mathematical Statements into Predicate Logic

Example : Translate “The sum of two positive integers is always positive” into a logical expression.

Solution:

1. Rewrite the statement to make the implied quantifiers and domains explicit:
   “For every two integers, if these integers are both positive, then the sum of these integers is positive.”
2. Introduce the variables *x* and *y*, and specify the domain, to obtain:
   “For all positive integers *x* and *y*, *x* + *y* is positive.”
3. The result is:

$$\forall x \forall y((x > 0) \land (y > 0) \rightarrow (x + y > 0))$$

where the domain of both variables consists of all integers

Translating Nested Quantifiers into English (1)

Example 1: Translate the statement

$$\forall x\left(C(X) \vee \exists y\left(C(y) \wedge F(X, y)\right)\right)$$

where C(x) is "*x* has a computer," and *F*(*x*,*y*) is "*x* and *y* are friends," and the domain for both *x* and *y* consists of all students in your school.

Solution: Every student in your school has a computer or has a friend who has a computer.

Translating Nested Quantifiers into English (2)

Example 2: Translate the statement

$$\exists x\forall y\forall z\Big(\big(F(x,y)\wedge F(x,z)\wedge(y\neq z)\big)\rightarrow\neg F(y,z)\Big)$$

where F(a, b) means a and b are friends and the domain for x, y, and z consists of all students in your school

Solution: There is a student none of whose friends are also friends with each other.

Translating English into Logical Expressions Example

Example: Use quantifiers to express the statement "There is a woman who has taken a flight on every airline in the world."

Solution:

1. Let $P(w,f)$ be "w has taken f " and $Q(f,a)$ be "f is a flight on a ."

2. The domain of w is all women, the domain of f is all flights, and the domain of a is all airlines.

3. Then the statement can be expressed as:

$$\exists w \forall a \exists f \left(P\left(w, f\right) \wedge Q\left(f, a\right)\right)$$

Questions on Translation from English (1)

Choose the obvious predicates and express in predicate logic.

Example 1: "Brothers are siblings."

Solution: Let $B(x,y)$ denote that x is a brother of y and $S(x, y)$ denote that x is a sibling of y

$$\forall x\, \forall y\, (B(x,y) \rightarrow S(x,y))$$

Example 2: "Siblinghood is symmetric."

Solution: Let $S(x, y)$ denote that x is a sibling of y

$$\forall x\, \forall y\, (S(x,y) \rightarrow S(y,x))$$

Example 3: "Everybody loves somebody."

Solution: Let $L(x, y)$ denote that x loves y

$$\forall x\, \exists y\, L(x,y)$$

Questions on Translation from English (2)

Choose the obvious predicates and express in predicate logic.

Example 4: "There is someone who is loved by everyone."

Solution: Let L(x, y) denote that x loves y

$\exists y \forall x\, L(x,y)$

Example 5: "There is someone who loves someone."

Solution: Let L(x, y) denote that x loves y

$\exists x \exists y\, L(x,y)$

Example 6: "Everyone loves himself"

Solution: Let L(x, y) denote that x loves y

$\forall x\, L(x,x)$

Negating Nested Quantifiers

Example 1: Recall the logical expression developed a few slides back:

$$\exists w \forall a \exists f \left(P(w,f) \wedge Q(f,a)\right)$$

Part 1: Use quantifiers to express the statement that "There does not exist a woman who has taken a flight on every airline in the world."

Solution: $\neg \exists w \forall a \exists f \left(P(w,f) \wedge Q(f,a)\right)$

Part 2: Now use De Morgan's Laws to move the negation as far inwards as possible.

Solution:

1. $\neg \exists w \forall a \exists f \left(P(w,f) \wedge Q(f,a)\right)$
2. $\forall w \neg \forall a \exists f \left(P(w,f) \wedge Q(f,a)\right)$ by De Morgan's for ∃ — $\neg\exists x P(x) \equiv \forall x \neg P(x)$
3. $\forall w \exists a \neg \exists f \left(P(w,f) \wedge Q(f,a)\right)$ by De Morgan's for ∀ — $\neg\forall x P(x) \equiv \exists x \neg P(x)$
4. $\forall w \exists a \forall f \neg \left(P(w,f) \wedge Q(f,a)\right)$ by De Morgan's for ∃ — $\neg\exists x P(x) \equiv \forall x \neg P(x)$
5. $\forall w \exists a \forall f \left(\neg P(w,f) \vee \neg Q(f,a)\right)$ by De Morgan's for ∧. — $\neg(P(x) \wedge Q(x)) \equiv \neg P(x) \vee \neg Q(x)$

Part 3: Can you translate the result back into English?

Solution:

"For every woman there is an airline such that for all flights, this woman has not taken that flight or that flight is not on this airline"

Revisiting Limitations of Propositional Calculus

- There are limitations to argument validation using propositional calculus
- **Example 1**

  P: Every computer connected to the university network is functioning properly

  ----

  $\therefore^?$ Computer MATH3 is functioning properly
- **Example 2**

  P: Computer CS2 is under attack by an intruder

  $\therefore^?$ There is a computer on the university network that is under attack by an intruder.
- **Example 3**

  P_1: All men are mortal

  P_2: Socrates is a man

  $\therefore^?$ Socrates is mortal

Additional Rules of Inference for Quantified Statements (1)

TABLE 2 Rules of Inference for Quantified Statements.

Rule of Inference	*Name*
$\dfrac{\forall x P(x)}{\therefore\ P(c)}$	Universal instantiation
$\dfrac{P(c) \text{ for an arbitrary } c}{\therefore\ \forall x P(x)}$	Universal generalization
$\dfrac{\exists x P(x)}{\therefore\ P(c) \text{ for some element } c}$	Existential instantiation
$\dfrac{P(c) \text{ for some element } c}{\therefore\ \exists x P(x)}$	Existential generalization

Additional Rules of Inference for Quantified Statements (2)

Mnemonics

x: Unspecified member of the domain
c: Specific member of the domain

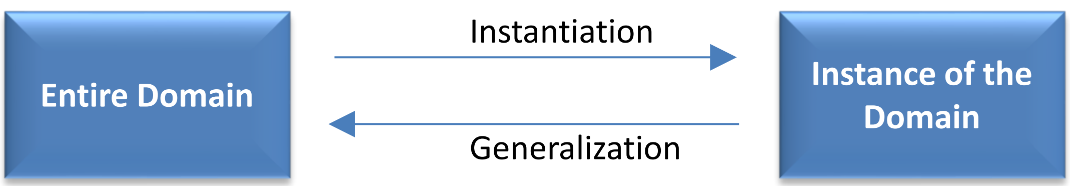


TABLE 2 **Rules of Inference for Quantified Statements.**

Rule of Inference	*Name*
$\forall x P(x)$ $\therefore P(c)$	Universal instantiation
$P(c)$ for an arbitrary c $\therefore \forall x P(x)$	Universal generalization
$\exists x P(x)$ $\therefore P(c)$ for some element c	Existential instantiation
$P(c)$ for some element c $\therefore \exists x P(x)$	Existential generalization

Universal Instantiation (UI)

$$\frac{\forall x P(x)}{\therefore P(c)}$$

Example:

Our domain consists of all dogs and Fido is a dog.

P: “All dogs are cuddly.”

C: “Therefore, Fido is cuddly.”

Universal Generalization (UG)

$$\frac{P(c) \text{ for an arbitrary } c}{\therefore \forall x P(x)}$$

P: Predicate is true for any arbitrarily selected member of the domain of discourse
----
C: ∴Predicate is universally true over the domain of discourse

By arbitrary we mean it is not any specific element of the domain and we cannot make any other assumption about *c* other than it comes from the domain.

UG is used often implicitly in Mathematical Proofs.

Existential Instantiation (EI)

$$\frac{\exists x P(x)}{\therefore P(c) \text{ for some element } c}$$

Example:

P: “There is someone who got an A in the course.”

[Upon verification we find that the student who got an A is named Michelle]

C: “Michelle got an A in the course”

Note that we are not selecting an arbitrary element from the domain. Rather we are selecting a specific element from the domain for which P(c) is true

Existential Generalization (EG)

$$\frac{P(c) \text{ for some element } c}{\therefore \exists x P(x)}$$

Example:

P: “Michelle got an A in the class.”

C: “Therefore, someone got an A in the class.”

Again we are not selecting an arbitrary element from the domain.