Discrete Structures

CSE 2315 (Spring 2014)

Lecture 6 Predicate and Quantifiers

Heng Huang, Ph.D.
Department of Computer Science and Engineering

Two more rules

- Deduction Method

  The argument $P_1 \wedge P_2 ,\ldots, P_n \rightarrow (R \rightarrow S)$ is tautologically equivalent to the argument $P_1 \wedge P_2 ,\ldots, P_n \wedge R \rightarrow S$.

- Example

  Prove that $[(A \rightarrow B) \wedge (B \rightarrow C)] \rightarrow (A \rightarrow C)$

- Proof

  Using the Deduction Method, the above argument can be rewritten as: $[(A \rightarrow B) \wedge (B \rightarrow C) \wedge A] \rightarrow C$. Now apply Modus Ponens twice!

- The rule $[(A \rightarrow B) \wedge (B \rightarrow C)] \rightarrow (A \rightarrow C)$ is called hypothesis syllogism and can be used directly.

One More Example

- Show that the argument $A' \wedge (B \rightarrow A) \rightarrow B'$ is valid.

- Is the following argument valid? “My client is left-handed, but if the diary is not missing, then my client is not left-handed; therefore, the diary is missing.”

- Caveat

  Lots of additional rules are provided on Page 33; you are not permitted to use any of them in your homework assignments, quizzes and exams! Derive them if you must use them.

Predicate Logic

- Propositional Logic has limited expressiveness. For instance, how would you capture the assertion, "Property P is true of every positive number"? $P_1 \wedge P_2$,…, P_∞ is neither compact nor useful.

- *Predicate logic* is an extension of propositional logic that permits concise reasoning about whole *classes* of entities.

  E.g., "x>1", "x+y=10"

- Such statements are neither true or false when the values of the variables are not specified.

Applications of Predicate Logic

- It is *the* formal notation for writing perfectly clear, concise, and unambiguous mathematical *definitions*, *axioms*, and *theorems* for *any* branch of mathematics.
- Supported by some of the more sophisticated *database query engines*.
- Basis for *automatic theorem provers* and many other Artificial Intelligence systems.

Subjects and Predicates

- The proposition

  "The dog is sleeping"

  has two parts:
  - "the dog" denotes the *subject* - the *object* or *entity* that the sentence is about.
  - "is sleeping" denotes the *predicate*- a property that the subject can have.

Propositional Functions

- A *predicate* is modeled as a *function* $P(\cdot)$ from objects to propositions.
  - $P(x)$ = "x is sleeping" (where x is any object).

- The *result of applying* a predicate P to an object x=a is the *proposition* $P(a)$.
  - *e.g.* if $P(x)$ = "$x > 1$",
    then $P(3)$ is the *proposition* "3 is greater than 1."

- <u>Note:</u> The predicate P **itself** (*e.g.* P="is sleeping") is **not** a proposition (not a complete sentence).

Quantifiers

- Basics
  - A collection of objects called domain of interpretation D.
  - Predicates are used to describe properties of objects. e.g., $P(x)$ could stand for the property that x is divisible by 3.
  - The universal quantifier $(\forall x)P(x)$ indicates that property P holds for all x in the domain D.
  - The existential quantifier $(\exists x)P(x)$ indicates that property P holds for some x in the domain D.

Quantifiers (contd.)

- The truth value of a predicate expression depends upon the domain of interpretation! For instance, consider the truth value of $(\forall x)P(x)$ in the following interpretations:
  - P(x) is the property that x is divisible by 2 and the domain D is the set of all even numbers.
  - P(x) is the property that x is divisible by 2 and the domain D is the set of all positive numbers.
- Predicates can be unary, binary, and in general, n-ary, e.g., P(x, y), Q(x1, x2, … , xn).
- The order of quantification is vital. For instance, $(\exists x)(\forall y)Q(x,y)$ does not mean the same thing as $(\forall y)(\exists x)Q(x,y)$.

Interpretation

- An interpretation of a predicate expression consists of the following:
  - A collection of objects called domain of interpretation, which must be non-empty.
  - An assignment of a property of the objects in the domain to each predicate in the expression.
  - An assignment of a particular object in the domain to each constant symbol in the expression.

- Example
  - What is the truth value of $(\forall x)P(x, a)$? The question is meaningless without the interpretation! Consider the following interpretation: The domain is the set of natural numbers $N = \{0, 1, \ldots\}$, $P(x, a)$ stands for $x >= a$ and a is 0. Clearly, in this interpretation the expression is true. What happens when the domain is the set of all integers?

Scope

- Bound and Free variables

  Consider the expression:

  $$(\forall x)[Q(x, y) \rightarrow (\exists y)R(x, y)]$$

  The x occurrences are bound to the $(\forall x)$ quantifier. The first y is said to be a free variable, since it is not bound to any quantifier. The scope of a quantifier is the portion of the predicate formula to which it applies.

Converting English to Predicate Logic

- Not an easy task! More than one result possible, depending on semantics of English language (which is not unambiguous). “Hang him not, let him free” and “Hang him, not let him free”!
- Example
  - All parrots are ugly.
  - Some parrots are ugly.
  - All dogs chase all rabbits.
  - Some dogs chase all rabbits.

Exercise

- Let S(x) denote "x is a student", I(x) denote "x is intelligent" and M(x) denote "x likes music". Give predicate wffs for:
  - All students are intelligent.
  - Some intelligent students like music.
  - Only intelligent students like music.

Validity

- A predicate wff is valid, if it is true in all interpretations. Validity in Predicate logic plays the role of tautology in propositional logic.

- Four fundamentally valid predicate wffs

(i) $(\forall x)P(x) \rightarrow (\exists x)P(x)$.

(ii) $(\forall x)P(x) \rightarrow P(a)$.

(iii) $(\forall x)(P(x) \wedge Q(x)) \rightarrow [(\forall x)P(x) \wedge (\forall x)Q(x)]$.

(iv) $P(x) \rightarrow [Q(x) \rightarrow P(x)]$.