Formal Method in Software Engineering (SE-313)

Course Teacher
Assistant Professor
Engr. Mustafa Latif

Predicate logic

Syntax of Predicate Logic

- **Two key types in predicate logic: terms and formulas**
- **Terms**
- **Basic components of predicate logic are called terms**
- **Any constant is a term.**
  - **We often use constants in math; we introduce them by writing things like Let S be the set {1; 2; 3}.**
  - **In this case, we have introduced a constant and made its denotation clear; we have given it an interpretation.**
  - **Definition: A constant of type T is a name that denotes some particular object in the set T.**

Predicate logic

Syntax of Predicate Logic

- **Any variable is a term.**
  - **A variable can stand for anything in a set of objects.**
  - **That is, a variable of type ℕ could stand for any of the natural numbers.**
  - **The variable x may stand for one of the days. We may let x = Monday, x = Tuesday, etc.**
  - **A collection of objects is called the domain of objects.**
- **Definition: A variable of type T is a name that can denote any value in the set T.**

3

Predicate logic

Syntax of Predicate Logic

- **Any function is a term.**
  - **The idea of functional terms in logic is similar to the idea of a function in programming: recall that in programming, a function is a procedure that takes some arguments, and returns a value.**
    - **PROCEDURE $f(a_1:T_1;\ \ldots\ a_n:T_n) : T$; this function takes n arguments; the first is of type T_1, the second is of type T_2, and so on. The function returns a value of type T.**
    - **Each symbol corresponds to a particular function.**
    - **Each function symbol is associated with a natural number called its arity. This is just the number of arguments it takes.**
- **Definition: Let f be an arbitrary function symbol of type T, with arity $n \in \mathbb{N}$, taking arguments of type $T_1; : : : ; T_n$ respectively. Also, let $\tau_1; : : : ; \tau_n$ be terms of type $T_1; : : : ; T_n$ respectively. Then $f(\tau_1; : : : ; \tau_n)$ is a functional term.**

4

Predicate logic

Syntax of Predicate Logic

Formula

- A well-formed formula in predicate logic is defined as
- If P is an n-ary predicate symbol and $t_1,\ldots,t_n$ are terms, then $P(t_1,\ldots t_n)$ is a formula. P, P(x,y,z)
- If the equality symbol (=) is considered part of logic and t_1 and t_2 are terms, then $t_1=t_2$ is a formula. P(a,b)=P(c,d)
- If φ is a formula asnd x isa a variable, then $(\forall x\varphi)$ and $(\exists x\varphi)$ are formulas. $[(\forall x)A(x)]' \Leftrightarrow (\exists x)[A(x)]'$

5

Predicate logic

Quantifiers

- Propositional wffs have rather limited expressive power. E.g., "For every x, x > 0".
- Quantifiers: Quantifiers are phrases that refer to given quantities, such as "for some" or "for all" or "for every", indicating how many objects have a certain property.
- Two kinds of quantifiers:
- Universal Quantifier: represented by ∀, "for all", "for every", "for each", or "for any".
- Existential Quantifier: represented by ∃, "for some", "there exists", "there is a", or "for at least one".

6

Predicate logic

Predicates

- Predicate: It is the verbal statement which describes the property of a variable. Usually represented by the letter P, the notation P(x) is used to represent some unspecified property or predicate that x may have.
  - P(x) = x has 30 days.
  - P(April) = April has 30 days.
  - What is the truth value of $(\forall x)P(x)$ where x is all the months and P(x) = x has less than 32 days
- Combining the quantifier and the predicate, we get a complete statement of the form $(\forall x)P(x)$ or $(\exists x)P(x)$
- The collection of objects is called the domain of interpretation, and it must contain at least one object.

7

Predicate logic

Unary, Binary,…, N-ary Predicates

- Predicates involving properties of a single variable: unary predicates
- Binary, ternary and n-ary predicates are also possible
- $(\forall x)\ (\exists y)Q(x,y)$ is a binary predicate. This expression reads as "for every x there exists a y such that Q(x,y)"
- Constants are also allowed in expressions, such as a, b, c, 0, 1, 2, etc.

8

Predicate logic

Interpretation

- An interpretation for an expression involving predicates consists of the following:
  - A collection of objects, called domain of interpretation, which must include at least one object.
  - 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.
- Predicate wffs can be built similar to propositional wffs using logical connectives with predicates and quantifiers.
- Must obey the rules of syntax to be considered a wff
- Examples of predicate wffs
  - $(\forall x)[P(x) \rightarrow Q(x)]$
  - $(\forall x)\ ((\exists\ y)[P(x,y) \vee Q(x,y)] \rightarrow R(x))$
  - $S(x,y) \wedge R(x,y)$

9

Predicate logic

Scope of a Variable in an Expression

- The parentheses or brackets are used wisely to identify the scope of the variable:
  - $(\forall x)\ ((\exists\ y)[P(x,y) \vee Q(x,y)] \longrightarrow R(x))$
    - Scope of $(\exists\ y)$ is $P(x,y) \vee Q(x,y)$ while the scope of $(\forall x)$ is the entire expression
  - $(\forall x)S(x) \vee (\exists y)R(y)$
    - Scope of $(\forall x)$ is $S(x)$ while the scope of $(\exists y)$ is $R(y)$
  - $(\forall x)[P(x,y) \longrightarrow (\exists\ y)\ Q(x,y)]$
    - Scope of variable y is not defined for P(x,y) hence y is called a free variable. Such expressions might not have a truth value at all.
    - $P(x)$: $x > 0$; $P(y)$^ $P(5)$, $P(y) \vee P(5)$.
- What is the truth of the wff $(\exists\ x)(A(x) \wedge (\forall\ y)[B(x,y) \longrightarrow C(y)])$, where A(x) is "$x > 0$" , B(x, y) is "$x > y$", C(y) is "$y \leq 0$", and x is the domain of positive integers and y is the domain of all integers?

10

Translation of Verbal Statements to Symbolic Form Using Intermediate Statements

- "Every person is nice" can be rephrased as "For any thing, if it is a person, then it is nice". So, if P(x) is "x is a person" and Q(x) be "x is nice", the statement can be symbolized as
  - $(\forall x)[P(x) \longrightarrow Q(x)]$
- Variations: "All persons are nice" or "Each person is nice".
- "There is a nice person" can be rewritten as "There exists something that is both a person and nice" in symbolic form
  - $(\exists x)[P(x) \land Q(x)]$
- Variations: "Some persons are nice" or "There are nice persons"
- So almost always, ∃ goes with Λ (conjunction) and ∀ goes with ⟶ (implication)

11

Translation of Verbal Statements to Symbolic

- Example for forming symbolic forms from predicate symbols:
  - D(x) is "x is dog"
  - R(x) is "x is a rabbit"
  - C(x,y) is "x chases y"
- All dogs chase all rabbits ⇔
  - For anything, if it is a dog, then for any other thing, if it is a rabbit, then the dog chases it ⇔ $(\forall x)[D(x) \longrightarrow (\forall y)(R(y) \rightarrow C(x,y))]$
- Some dogs chase all rabbits ⇔
  - There is something that is a dog and for any other thing, if that thing is a rabbit, then the dog chases it ⇔ $(\exists x)[D(x) \land (\forall y)(R(y) \longrightarrow C(x,y))]$
- Only dogs chase rabbits ⇔
  - For anything, if it is a rabbit then, if anything chases it, that thing is a dog ⇔ $(\forall y)\ [R(y) \longrightarrow (\forall x)\ (C(x, y) \longrightarrow D(x))]$
  - Or, for any two things, if one is a rabbit and the other chases it, then the other is a dog ⇔ $(\forall y)\ (\forall x)[R(y) \land C(x,y) \longrightarrow D(x)]$

12

Negation of Statements

- A(x): Everything is beautiful
  - Negation will be "it is false that everything is beautiful", i.e. "something is not beautiful"
  - In symbolic form, $[(\forall x)A(x)]' \Leftrightarrow (\exists x)[A(x)]'$
  - Similarly, negation of "Something is beautiful" is "Nothing is beautiful" or "Nothing is beautiful"
  - Hence, $[((\exists x)A(x)]' \Leftrightarrow (\forall x)[A(x)]'$

13

More Examples on Negation

- Some pictures are old and faded.
  - Every picture is not old or not faded.
- All people are tall and thin.
  - Someone is short or fat.
- Some students eat only pizza.
  - Every student eats something which is not a pizza
- Only students eat pizza.
  - There is a non-student who eats pizza.

14

- S(x): x is a student
- I(x): x is intelligent
- M(x): x likes music

- Write wffs than express the following statements:
  - All students are intelligent.
  - Some intelligent students like music.
  - Everyone who likes music is a stupid student.
  - Only intelligent students like music.

For anything, if it is a student, then it is intelligent $\Leftrightarrow$ $(\forall x)[S(x) \rightarrow I(x)]$

There is something that is intelligent and it is a student and it likes music $\Leftrightarrow$ $(\exists x)[I(x) \wedge S(x) \wedge M(x)]$

For anything, if that thing likes music, then it is a student and it is not intelligent $\Leftrightarrow$

$(\forall x)(M(x) \rightarrow S(x) \wedge [I(x)]')$

For any thing, if it likes music, then it is a student and it is intelligent $\Leftrightarrow$

$(\forall x)(M(x) \rightarrow S(x) \wedge I(x))$