Predicate Logic

Predicate Logic

- *Predicate logic* is an extension of propositional logic that permits concisely reasoning about whole *classes* of entities.
- Propositional logic treats simple *propositions* (sentences) as atomic entities.
- In contrast, *predicate* logic distinguishes the *subject* of a sentence from its *predicate.*
  - Remember these English grammar terms?

Other Applications

- Predicate logic is the foundation of the field of *mathematical logic*, which culminated in *Gödel's incompleteness theorem*, which revealed the ultimate limits of mathematical thought:
  - Given any finitely describable, consistent proof procedure, there will still be *some* true statements that can *never be proven* by that procedure.
- *I.e.*, we can't discover *all* mathematical truths, unless we sometimes resort to making *guesses.*

Practical Applications

- Basis for clearly expressed formal specifications for any complex system.
- Basis for *automatic theorem provers* and many other Artificial Intelligence systems.
- Supported by some of the more sophisticated *database query engines* and *container class libraries* (these are types of programming tools).

More About Predicates

- Convention: Lowercase variables x, y, z... denote objects/entities; uppercase variables P, Q, R… denote propositional functions (predicates).
- Keep in mind that the *result of applying* a predicate P to an object x is the *proposition* $P(x)$. But the predicate P **itself** (*e.g.* P="is sleeping") is **not** a proposition (not a complete sentence).
  - *E.g.* if $P(x)$ = "x is a prime number", $P(3)$ is the *proposition* "3 is a prime number."

Propositional Functions

- Predicate logic *generalizes* the grammatical notion of a predicate to also include propositional functions of **any** number of arguments, each of which may take **any** grammatical role that a noun can take.
  - *E.g.* let $P(x,y,z)$ = "x gave y the grade z", then if x="Mike", y="Mary", z="A", then $P(x,y,z)$ = "Mike gave Mary the grade A."

Quantifier Expressions

- *Quantifiers* provide a notation that allows us to *quantify* (count) *how many* objects in the universe of disclosure satisfy a given predicate.
- "∀" is the FOR∀LL or *universal* quantifier. $\forall x\, P(x)$ means *for all* x in the u.d., *P* holds.
- "∃" is the ∃XISTS or *existential* quantifier. $\exists x\, P(x)$ means there *exists* an *x* in the u.d. (that is, 1 or more) such that $P(x)$ is true.

Free and Bound Variables

- An expression like $P(x)$ is said to have a *free variable* x (meaning, x is undefined).
- A quantifier (either $\forall$ or $\exists$) *operates* on an expression having one or more free variables, and *binds* one or more of those variables, to produce an expression having one or more *bound variables*.

Example of Binding

- $P(x,y)$ has 2 free variables, x and y.
- $\forall x\, P(x,y)$ has 1 free variable, and one bound variable. [Which is which?]
- "$P(x)$, where $x=3$" is another way to bind x.
- An expression with zero free variables is a bona-fide (actual) proposition.
- An expression with one or more free variables is still only a predicate: $\forall x\, P(x,y)$

Quantifier Equivalence Laws

- Definitions of quantifiers: If u.d.=a,b,c,…
  $\forall x\, P(x) \Leftrightarrow P(\mathrm{a}) \wedge P(\mathrm{b}) \wedge P(\mathrm{c}) \wedge \ldots$
  $\exists x\, P(x) \Leftrightarrow P(\mathrm{a}) \vee P(\mathrm{b}) \vee P(\mathrm{c}) \vee \ldots$
- From those, we can prove the laws:
  $\forall x\, P(x) \Leftrightarrow \neg\exists x\, \neg P(x)$
  $\exists x\, P(x) \Leftrightarrow \neg\forall x\, \neg P(x)$
- Which *propositional* equivalence laws can be used to prove this?

DeMorgan's

Review: Predicate Logic

- Objects x, y, z, …
- Predicates P, Q, R, … are functions mapping objects x to propositions $P(x)$.
- Multi-argument predicates $P(x, y)$.
- Quantifiers: $[\forall x\, P(x)]$:≡ "For all x's, $P(x)$."
  $[\exists x\, P(x)]$:≡ "There is an x such that $P(x)$."
- Universes of discourse, bound & free vars.

Nesting of Quantifiers

Example: Let the u.d. of x & y be people.

Let $L(x,y)$="x likes y" (a predicate w. 2 f.v.'s)

Then $\exists y\, L(x,y)$ = "There is someone whom x likes." (A predicate w. 1 free variable, x)

Then $\forall x\, (\exists y\, L(x,y))$ = "Everyone has someone whom they like."

(A <u>Proposition</u> with <u>0</u> free variables.)

Quantifier Exercise

If $R(x,y)$=“x relies upon y,” express the following in unambiguous English:

$\forall x(\exists y\, R(x,y))=$ Everyone has *someone* to rely on.

$\exists y(\forall x\, R(x,y))=$ There’s a poor overburdened soul whom *everyone* relies upon (including himself)!

$\exists x(\forall y\, R(x,y))=$ There’s some needy person who relies upon *everybody* (including himself).

$\forall y(\exists x\, R(x,y))=$ Everyone has *someone* who relies upon them.

$\forall x(\forall y\, R(x,y))=$ *Everyone* relies upon *everybody*, (including themselves)!

Natural language is ambiguous!

- "Everybody likes somebody."
  - For everybody, there is somebody they like,
    - $\forall x \exists y\ Likes(x,y)$ [Probably more likely.]
  - or, there is somebody (a popular person) whom everyone likes?
    - $\exists y \forall x\ Likes(x,y)$
- "Somebody likes everybody."
  - Same problem: Depends on context, emphasis.

More to Know About Binding

- $\forall x \exists x P(x)$ - x is not a free variable in $\exists x P(x)$, therefore the $\forall x$ binding <u>isn't used</u>.
- $(\forall x P(x)) \wedge Q(x)$ - The variable x is outside of the *scope* of the $\forall x$ quantifier, and is therefore free. Not a complete proposition!
- $(\forall x P(x)) \wedge (\exists x Q(x))$ – This is legal, because there are 2 <u>different</u> x's!

More Equivalence Laws

- $\forall x\, \forall y\, P(x,y) \Leftrightarrow \forall y\, \forall x\, P(x,y)$
- $\exists x\, \exists y\, P(x,y) \Leftrightarrow \exists y\, \exists x\, P(x,y)$
- $\forall x\, \mathbf{(}P(x) \wedge Q(x)\mathbf{)} \Leftrightarrow \mathbf{(}\forall x\, P(x)\mathbf{)} \wedge \mathbf{(}\forall x\, Q(x)\mathbf{)}$
- $\exists x\, \mathbf{(}P(x) \vee Q(x)\mathbf{)} \Leftrightarrow \mathbf{(}\exists x\, P(x)\mathbf{)} \vee \mathbf{(}\exists x\, Q(x)\mathbf{)}$

Calculus Example

- One way of precisely defining the calculus concept of a *limit*, using quantifiers:

$$\left(\lim_{x\to a} f(x) = L\right) \Leftrightarrow$$

$$\left(\begin{array}{l}\forall \varepsilon > 0 : \exists \delta > 0 : \forall x : \\ \left(|x-a| < \delta\right) \rightarrow \left(|f(x) - L| < \varepsilon\right)\end{array}\right)$$

Example

- Definitions: $H(x) :\equiv$ "x is human"; $M(x) :\equiv$ "x is mortal"; $G(x) :\equiv$ "x is a god"
- Premises:
  - $\forall x\, H(x) \rightarrow M(x)$ ("Humans are mortal") and
  - $\forall x\, G(x) \rightarrow \neg M(x)$ ("Gods are immortal").
- Show that $\neg\exists x\, (H(x) \wedge G(x))$ ("No human is a god.")

The Derivation

- $\forall x\, H(x) \rightarrow M(x)$ and $\forall x\, G(x) \rightarrow \neg M(x)$.
- $\forall x\, \neg M(x) \rightarrow \neg H(x)$ **[Contrapositive.]**
- $\forall x\, [G(x) \rightarrow \neg M(x)] \wedge [\neg M(x) \rightarrow \neg H(x)]$
- $\forall x\, G(x) \rightarrow \neg H(x)$ **[Transitivity of $\rightarrow$.]**
- $\forall x\, \neg G(x) \vee \neg H(x)$ **[Definition of $\rightarrow$.]**
- $\forall x\, \neg(G(x) \wedge H(x))$ **[DeMorgan's law.]**
- $\neg \exists x\, G(x) \wedge H(x)$ **[An equivalence law.]**

Predicate Logic

- From these sections you should have learned:
  - Predicate logic notation & conventions
  - Conversions: predicate logic $\leftrightarrow$ clear English
  - Meaning of quantifiers, equivalences
  - Simple reasoning with quantifiers