Logica del Primo Ordine 1
First-Order Logic 1

Wolfgang Faber

University of Calabria, Italy

2007

Outline

Propositional Logic $\Rightarrow$ Done?

We want to represent:

- Socrates is a human.
- Humans are mortal.

From this, we want to draw the conclusion:

- Socrates is mortal.

Propositional Logic $\Rightarrow$ Done?

We want to represent:

- Socrates is a human.
- Humans are mortal.

From this, we want to draw the conclusion:

- Socrates is mortal.

Propositional Logic ⇒ Done?

In propositional logic:

- variable *SH* (Socrates is a human.)
- variable *HM* (Humans are mortal.)
- variable *SM* (Socrates is mortal.)
- formula *F*: $SH \wedge$ (“Socrates is a human” and ...)
- $SH \rightarrow SM$ (“Socrates is a human” implies “Socrates is mortal.”)

From this, we can draw the conclusion:

- $F \models SM$ (“Socrates is mortal.”)
- ...and where is *HM*??
- This is not what we wanted to express!

Propositional Logic ⇒ Done?

In propositional logic:

- variable *SH* (Socrates is a human.)
- variable *HM* (Humans are mortal.)
- variable *SM* (Socrates is mortal.)
- formula F: $SH \wedge$ (“Socrates is a human” and . . .)
- $SH \rightarrow SM$ (“Socrates is a human” implies “Socrates is mortal.”)

From this, we can draw the conclusion:

- $F \models SM$ (“Socrates is mortal.”)
- . . . and where is *HM*??
- This is not what we wanted to express!

Propositional Logic $\Rightarrow$ Done?

In propositional logic:

- variable SH (Socrates is a human.)
- variable HM (Humans are mortal.)
- variable SM (Socrates is mortal.)
- formula F: $SH \wedge$ ("Socrates is a human" and ...)
- $SH \rightarrow SM$ ("Socrates is a human" implies "Socrates is mortal.")

From this, we can draw the conclusion:

- $F \models SM$ ("Socrates is mortal.")
- ...and where is HM??
- This is not what we wanted to express!

Propositional Logic $\Rightarrow$ Done?

In propositional logic:

- variable *SH* (Socrates is a human.)
- variable *HM* (Humans are mortal.)
- variable *SM* (Socrates is mortal.)
- formula *F*: $SH \wedge$ (“Socrates is a human” and ...)
- $SH \rightarrow SM$ (“Socrates is a human” implies “Socrates is mortal.”)

From this, we can draw the conclusion:

- $F \models SM$ (“Socrates is mortal.”)
- ...and where is *HM*??
- This is not what we wanted to express!

Propositional Logic ⇒ Done?

“Humans are mortal” is not an atom!

- Marco is a human.
- Humans are mortal.
- Marco is mortal.

- We talk about “objects,” not about propositions!
- But: There are no objects in propositional logic.

Propositional Logic ⇒ Done?

“Humans are mortal” is not an atom!

- Marco is a human.
- Humans are mortal.
- Marco is mortal.

- We talk about “objects,” not about propositions!
- But: There are no objects in propositional logic.

Propositional Logic ⇒ Done?

"Humans are mortal" is not an atom!

- Marco is a human.
- Humans are mortal.
- Marco is mortal.

- We talk about "objects," not about propositions!
- But: There are no objects in propositional logic.

Propositional Logic ⇒ Done?

"Humans are mortal" is not an atom!

- Marco is a human.
- Humans are mortal.
- Marco is mortal.

- We talk about "objects," not about propositions!
- But: There are no objects in propositional logic.

Outline

Towards First-Order Logic

- “Start” from Propositional Logic.
- “Atoms” are no longer indivisible.
- Compose “atoms” from:
  - terms (representing objects) and
  - predicates (statements about terms).
- Socrates $\Rightarrow$ term
- Mortal $\Rightarrow$ predicate

Towards First-Order Logic

- "Start" from Propositional Logic.
- "Atoms" are no longer indivisible.
- Compose "atoms" from:
  - terms (representing objects) and
  - predicates (statements about terms).
- Socrates $\Rightarrow$ term
- Mortal $\Rightarrow$ predicate

Towards First-Order Logic

- “Start” from Propositional Logic.
- “Atoms” are no longer indivisible.
- Compose “atoms” from:
  - terms (representing objects) and
  - predicates (statements about terms).
- Socrates $\Rightarrow$ term
- Mortal $\Rightarrow$ predicate

Towards First-Order Logic

- “Start” from Propositional Logic.
- “Atoms” are no longer indivisible.
- Compose “atoms” from:
  - terms (representing objects) and
  - predicates (statements about terms).
- Socrates $\Rightarrow$ term
- Mortal $\Rightarrow$ predicate

Functions

- “Socrates’ father”
- . . . represents exactly one “object,”
- but we refer to it from another “object.”
- Function symbols map some “objects” to another “object”.
- “Objects” are constant function symbols!

Functions

- “Socrates’ father”
- . . . represents exactly one “object,”
- but we refer to it from another “object.”
- Function symbols map some “objects” to another “object”.
- “Objects” are constant function symbols!

Functions

- “Socrates’ father”
- . . . represents exactly one “object,”
- but we refer to it from another “object.”
- Function symbols map some “objects” to another “object”.
- “Objects” are constant function symbols!

Functions

- “Socrates’ father”
- . . . represents exactly one “object,”
- but we refer to it from another “object.”
- Function symbols map some “objects” to another “object”.
- “Objects” are constant function symbols!

Variable

- "Humans are mortal"
- ...if some "object" is human, it is mortal.
- ...We need a variable ranging over "objects."
- Note: Such variables are completely different from propositional variables!
- Think about them as object variables.

Variable

- "Humans are mortal"
- . . . if some "object" is human, it is mortal.
- . . . We need a variable ranging over "objects."
- Note: Such variables are completely different from propositional variables!
- Think about them as object variables.

Variable

- "Humans are mortal"
- . . . if some "object" is human, it is mortal.
- . . . We need a variable ranging over "objects."
- Note: Such variables are completely different from propositional variables!
- Think about them as object variables.

Variable

- "Humans are mortal"
- . . . if some "object" is human, it is mortal.
- . . . We need a variable ranging over "objects."
- Note: Such variables are completely different from propositional variables!
- Think about them as object variables.

Quantifiers

- “Humans are mortal”
- . . . actually wants to express “all humans are mortal,”
- . . . and not “some humans are mortal,”
- . . . and certainly not “no humans are mortal.”
- Quantifiers express
  1. “for all objects” or
  2. “some object exists.”

Quantifiers

- “Humans are mortal”
- . . . actually wants to express “all humans are mortal,”
- . . . and not “some humans are mortal,”
- . . . and certainly not “no humans are mortal.”
- Quantifiers express
  1. “for all objects” or
  2. “some object exists.”

Quantifiers

- "Humans are mortal"
- . . . actually wants to express "all humans are mortal,"
- . . . and not "some humans are mortal,"
- . . . and certainly not "no humans are mortal."
- Quantifiers express
  1. "for all objects" or
  2. "some object exists."

Quantifiers

- "Humans are mortal"
- ...actually wants to express "all humans are mortal,"
- ...and not "some humans are mortal,"
- ...and certainly not "no humans are mortal."
- Quantifiers express
  1. "for all objects" or
  2. "some object exists."

Quantifiers

- “Humans are mortal”
- . . . actually wants to express “all humans are mortal,”
- . . . and not “some humans are mortal,”
- . . . and certainly not “no humans are mortal.”
- Quantifiers express
  1. “for all objects” or
  2. “some object exists.”

Outline

Function Symbols, Constants

- Countable set F of function symbols.
- An arity (nonnegative integer) is associated with each function symbol.
- Function symbols with arity 0 are constants (“objects”).
- Examples:
  - *socrates* (arity 0)
  - *father* (arity 1)
  - *son* (arity 2)
  - *supercalifragilistichespiralidoso*287 (arity 6)

Function Symbols, Constants

- Countable set *F* of function symbols.
- An arity (nonnegative integer) is associated with each function symbol.
- Function symbols with arity 0 are constants (“objects”).
- Examples:
  - *socrates* (arity 0)
  - *father* (arity 1)
  - *son* (arity 2)
  - *supercalifragilistichespiralidoso*287 (arity 6)

Function Symbols, Constants

- Countable set F of function symbols.
- An arity (nonnegative integer) is associated with each function symbol.
- Function symbols with arity 0 are constants (“objects”).
- Examples:
  - *socrates* (arity 0)
  - *father* (arity 1)
  - *son* (arity 2)
  - *supercalifragilistichespiralidoso*287 (arity 6)

Function Symbols, Constants

- Countable set F of function symbols.
- An arity (nonnegative integer) is associated with each function symbol.
- Function symbols with arity 0 are constants (“objects”).
- Examples:
  - *socrates* (arity 0)
  - *father* (arity 1)
  - *son* (arity 2)
  - *supercalifragilistichespiralidoso*287 (arity 6)

Function Symbols, Constants

- Countable set F of function symbols.
- An arity (nonnegative integer) is associated with each function symbol.
- Function symbols with arity 0 are constants (“objects”).
- Examples:
  - *socrates* (arity 0)
  - *father* (arity 1)
  - *son* (arity 2)
  - *supercalifragilistichespiralidoso*287 (arity 6)

Variables

- Countable set V of (object) variables.
- V and F are disjoint!
- Examples:
  - *Human*
  - *Xiknve*
  - A, B, C, D

Variables

- Countable set V of (object) variables.
- V and F are disjoint!
- Examples:
  - $Human$
  - $Xiknve$
  - A, B, C, D

Terms

Build terms from function symbols and variables,respecting arities.
Again: Inductive definition.

- If v is a variable symbol, v is a term.
- If f is a function symbol of arity 0, f is a term.
- If f is a function symbol of arity $n > 0$, and $t_1, \dots, t_n$ are terms, then $f(t_1, \dots, t_n)$ is a term.

Ground term: Term not containing any variable.

Terms

Build terms from function symbols and variables,respecting arities.
Again: Inductive definition.

- If v is a variable symbol, v is a term.
- If f is a function symbol of arity 0, f is a term.
- If f is a function symbol of arity $n > 0$, and $t_1, \ldots, t_n$ are terms, then $f(t_1, \ldots, t_n)$ is a term.

Ground term: Term not containing any variable.

Terms

Build terms from function symbols and variables,respecting arities.

Again: Inductive definition.

- If v is a variable symbol, v is a term.
- If f is a function symbol of arity 0, f is a term.
- If f is a function symbol of arity $n > 0$, and $t_1, \ldots, t_n$ are terms, then $f(t_1, \ldots, t_n)$ is a term.

Ground term: Term not containing any variable.

Terms

Build terms from function symbols and variables,respecting arities.
Again: Inductive definition.

- If v is a variable symbol, v is a term.
- If f is a function symbol of arity 0, f is a term.
- If f is a function symbol of arity $n > 0$, and $t_1, \ldots, t_n$ are terms, then $f(t_1, \ldots, t_n)$ is a term.

Ground term: Term not containing any variable.

Terms

Build terms from function symbols and variables,respecting arities.

Again: Inductive definition.

- If v is a variable symbol, v is a term.
- If f is a function symbol of arity 0, f is a term.
- If f is a function symbol of arity $n > 0$, and $t_1, \ldots, t_n$ are terms, then $f(t_1, \ldots, t_n)$ is a term.

Ground term: Term not containing any variable.

Terms – Examples

Example terms:

- $socrates$
- $Xiknve$
- $father(socrates)$
- $father(father(socrates))$
- $father(father(father(socrates)))$
- $son(father(socrates), Xiknve)$
- $supercalifragilistichespiralidoso287(A, B, C, socrates, Xiknve, D)$

Note that each term will represent an “object.”

Terms – Examples

Example terms:

- $socrates$
- $Xiknve$
- $father(socrates)$
- $father(father(socrates))$
- $father(father(father(socrates)))$
- $son(father(socrates), Xiknve)$
- $supercalifragilistichespiralidoso287(A, B, C, socrates, Xiknve, D)$

Note that each term will represent an "object."

Outline

Predicate Symbols

- Countable set P of predicate symbols.
- An arity is associated with each predicate symbol.
- Examples:
  - *human* (arity 1)
  - *mortal* (arity 1)
  - *married* (arity 2)
  - *yggdrasil* (arity 18)

Predicate Symbols

- Countable set P of predicate symbols.
- An arity is associated with each predicate symbol.
- Examples:
  - *human* (arity 1)
  - *mortal* (arity 1)
  - *married* (arity 2)
  - *yggdrasil* (arity 18)

Predicate Symbols

- Countable set P of predicate symbols.
- An arity is associated with each predicate symbol.
- Examples:
  - *human* (arity 1)
  - *mortal* (arity 1)
  - *married* (arity 2)
  - *yggdrasil* (arity 18)

Atoms

- If p is a predicate symbol of arity n, and
- $t_1, \ldots, t_n$ are terms, then
- $p(t_1, \ldots, t_n)$ is an atomic formula or atom.
- Examples:
  - $human(socrates)$
  - $mortal(father(Xiknve))$
  - $married(socrates, A)$

Predicates with arity 0 are like propositional atoms.

Atoms

- If p is a predicate symbol of arity n, and
- $t_1, \ldots, t_n$ are terms, then
- $p(t_1, \ldots, t_n)$ is an atomic formula or atom.
- Examples:
  - $human(socrates)$
  - $mortal(father(Xiknve))$
  - $married(socrates, A)$

Predicates with arity 0 are like propositional atoms.

Atoms

- If p is a predicate symbol of arity n, and
- $t_1, \ldots, t_n$ are terms, then
- $p(t_1, \ldots, t_n)$ is an atomic formula or atom.
- Examples:
  - $human(socrates)$
  - $mortal(father(Xiknve))$
  - $married(socrates, A)$

Predicates with arity 0 are like propositional atoms.

Formulas

Similar to propositional logic with different atoms.

x is a formula if and only if

- x is an atomic formula, or
- $x = \top$, or
- $x = \bot$, or
- $x = (\neg y)$ where y is a formula, or
- $x = (y \wedge z)$ where y, z are formulas, or
- $x = (y \vee z)$ where y, z are formulas, or
- $x = (y \rightarrow z)$ where y, z are formulas, or
- $x = (y \leftrightarrow z)$ where y, z are formulas, or
- $x = (\forall V\ y)$ where V is a variable and y is a formula, or
- $x = (\exists V\ y)$ where V is a variable and y is a formula.

Formulas

Similar to propositional logic with different atoms.
x is a formula if and only if

- x is an atomic formula, or
- $x = \top$, or
- $x = \bot$, or
- $x = (\neg y)$ where y is a formula, or
- $x = (y \wedge z)$ where y, z are formulas, or
- $x = (y \vee z)$ where y, z are formulas, or
- $x = (y \rightarrow z)$ where y, z are formulas, or
- $x = (y \leftrightarrow z)$ where y, z are formulas, or
- $x = (\forall V\ y)$ where V is a variable and y is a formula, or
- $x = (\exists V\ y)$ where V is a variable and y is a formula.

Quantified Variables

- $\forall$ is the universal quantifier.
- $\exists$ is the existential quantifier.
- In $\forall V\ y$ and $\exists V\ y$, y is the scope of V.
- In $\forall V\ y$ and $\exists V\ y$, V is bound in y.
- Variables which are not bound in a formula are free.
- Formulas without free variables are closed or sentences.

Formulas: Examples

As for propositional logic, we minimize parentheses.

- $\mathit{human}(\mathit{socrates})$
- $\mathit{mortal}(\mathit{socrates})$
- $\forall A\ (\mathit{human}(A) \rightarrow \mathit{mortal}(A))$
- $\forall A\ (\mathit{human}(\mathit{son}(\mathit{socrates}, A)) \rightarrow \mathit{married}(\mathit{socrates}, A))$
- $\forall A\ ((\exists B\ \mathit{human}(\mathit{son}(A, B))) \rightarrow (\exists C\ \mathit{married}(A, C)))$
- $((\exists B\ \mathit{human}(\mathit{son}(A, B))) \rightarrow (\exists C\ \mathit{married}(A, C)))$

Practical Example 1

$$n(z)$$
$$\forall X\ (n(X) \rightarrow n(s(X)))$$
$$\forall X\ \forall Y\ (\neg(e(X, Y)) \rightarrow \neg(e(s(X), s(Y))))$$
$$\forall X\ \neg e(s(X), z)$$

Practical Example 2

$$\forall X\ \neg e(s(X), z)$$
$$\forall X\ \forall Y (e(s(X), s(Y)) \rightarrow e(X, Y))$$
$$\forall X\ e(a(X, z), X)$$
$$\forall X\ \forall Y\ e(a(X, s(Y)), s(a(X, Y)))$$
$$\forall X\ e(m(X, z), z)$$
$$\forall X\ \forall Y\ e(m(X, s(Y)), a(m(X, Y), X))$$

Definition of Free Variables

- Given a term t, the set of its free variables $free(t)$ is defined as:
  - $free(t) = \{X\}$ if t is a variable X,
  - $free(t) = \{\}$ if t is a constant,
  - $free(t) = \bigcup_{i=1}^{n} free(t_i)$ if t is $f(t_1, \ldots, t_n)$.
- Given a formula f, the set of its free variables $free(f)$ is defined as:
  - $free(f) = \bigcup_{i=1}^{n} free(t_i)$ if f is an atom $p(t_1, \ldots, t_n)$,
  - $free(f) = \{\}$ if f is $\top$ or $\bot$,
  - $free(f) = free(g)$ if f is $\neg g$,
  - $free(f) = free(g) \cup free(h)$ if f is $g \circ h$ for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$,
  - $free(f) = free(g) \setminus X$ if f is $\forall X\ g$ or $\exists X\ g$

Definition of Free Variables

- Given a term t, the set of its free variables $\mathit{free}(t)$ is defined as:
  - $\mathit{free}(t) = \{X\}$ if t is a variable X,
  - $\mathit{free}(t) = \{\}$ if t is a constant,
  - $\mathit{free}(t) = \bigcup_{i=1}^{n} \mathit{free}(t_i)$ if t is $f(t_1, \ldots, t_n)$.
- Given a formula f, the set of its free variables $\mathit{free}(f)$ is defined as:
  - $\mathit{free}(f) = \bigcup_{i=1}^{n} \mathit{free}(t_i)$ if f is an atom $p(t_1, \ldots, t_n)$,
  - $\mathit{free}(f) = \{\}$ if f is $\top$ or $\bot$,
  - $\mathit{free}(f) = \mathit{free}(g)$ if f is $\neg g$,
  - $\mathit{free}(f) = \mathit{free}(g) \cup \mathit{free}(h)$ if f is $g \circ h$ for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$,
  - $\mathit{free}(f) = \mathit{free}(g) \setminus X$ if f is $\forall X\ g$ or $\exists X\ g$

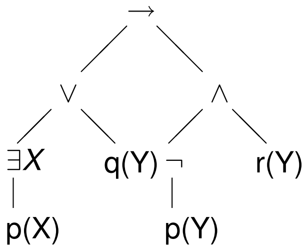
Formulas as Trees

Every formula can be written as a formula tree:

$\exists X\ p(X) \vee q(Y) \rightarrow \neg p(Y) \wedge r(Y)$

Formulas as Trees

Every formula can be written as a formula tree:
$\exists X\, p(X) \vee q(Y) \rightarrow \neg p(Y) \wedge r(Y)$



Term Substitution

- Substitute term x by y in term t, denoted $t[y/x]$:
  - $t[y/x] = y$ if $t = x$,
  - $t[y/x] = t$ if $t \neq x$,
  - $t[y/x] = f(t_1[y/x], \ldots, t_n[y/x])$ if $t \neq x$ and $t = f(t_1, \ldots, t_n)$.
- Substitute term x by y in a formula f, denoted $f[y/x]$:
  - $f[y/x] = f$ if f is $\top$ or $\bot$
  - $f[y/x] = p(t_1[y/x], \ldots, t_n[y/x])$ if f is an atom $p(t_1, \ldots, t_n)$
  - $f[y/x] = \neg g[y/x]$ if $f = \neg g$
  - $f[y/x] = g[y/x] \circ h[y/x]$ if $f = g \circ h$ for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$,
  - $f[y/x] = \mathcal{Q}V\ g[y/x]$ if $f = \mathcal{Q}V\ g$ for $\mathcal{Q} \in \{\forall, \exists\}$ and $V \neq x$,
  - $f[y/x] = \mathcal{Q}V\ g$ if $f = \mathcal{Q}V\ g$ for $\mathcal{Q} \in \{\forall, \exists\}$ and $V = x$.

Term Substitution

- Substitute term x by y in term t, denoted $t[y/x]$:
  - $t[y/x] = y$ if $t = x$,
  - $t[y/x] = t$ if $t \neq x$,
  - $t[y/x] = f(t_1[y/x], \ldots, t_n[y/x])$ if $t \neq x$ and $t = f(t_1, \ldots, t_n)$.
- Substitute term x by y in a formula f, denoted $f[y/x]$:
  - $f[y/x] = f$ if f is $\top$ or $\bot$
  - $f[y/x] = p(t_1[y/x], \ldots, t_n[y/x])$ if f is an atom $p(t_1, \ldots, t_n)$
  - $f[y/x] = \neg g[y/x]$ if $f = \neg g$
  - $f[y/x] = g[y/x] \circ h[y/x]$ if $f = g \circ h$ for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$,
  - $f[y/x] = \mathcal{Q}V\ g[y/x]$ if $f = \mathcal{Q}V\ g$ for $\mathcal{Q} \in \{\forall, \exists\}$ and $V \neq x$,
  - $f[y/x] = \mathcal{Q}V\ g$ if $f = \mathcal{Q}V\ g$ for $\mathcal{Q} \in \{\forall, \exists\}$ and $V = x$.

Formula Substitution

- Substitute a formula g by h in a formula f, denoted $f[h/g]$:
- $\Rightarrow$ As for propositional logic.

Outline

Semantics – Requirements

Give meaning to

- function symbols (including constants),
- variable symbols
- predicate symbols
- formulas

Domain

- We want to speak about “objects.”
- Each term represents an “object.”
- Domain $\mathcal{D}$: A set of “objects.”

Interpretation Function

- Interpretation I: Associate terms with objects.
- For each constant symbol c, $I(c) \in \mathcal{D}$
- For each function symbol f of arity n, $I(f)$ is a function $\mathcal{D}^n \mapsto \mathcal{D}$
- For any n-tuple of objects, $I(f)$ defines a unique object.

Interpretation Function

- Interpretation I: Also associate predicates with truth functions.
- For each constant predicate p of arity n, $I(p)$ is a function $\mathcal{D}^n \mapsto \{0, 1\}$
- For any n-tuple of objects, $I(p)$ defines a truth value.

Free Variables?

- There are two ways to handle free variables in formulas:
  1. Define variable assignments, associating a domain object with a variable.
  2. Consider universal closure: $\forall X_1 \ \dots \forall X_n \ f$ where $free(f) = \{X_1, \dots, X_n\}$.
- We will consider variable assignments:
- A variable assignment v is a function $V \mapsto \mathcal{D}$.

Free Variables?

- There are two ways to handle free variables in formulas:
  1. Define variable assignments, associating a domain object with a variable.
  2. Consider universal closure: $\forall X_1 \ \ldots \forall X_n \ f$ where $free(f) = \{X_1, \ldots, X_n\}$.
- We will consider variable assignments:
- A variable assignment v is a function $V \mapsto \mathcal{D}$.

Outline

Valuation

- A valuation μ_M of a term t with respect to a first order structure $M = (\mathcal{D}, I, v)$ is:
  - $\mu_M(t) = v(t)$ if t is a variable
  - $\mu_M(t) = I(t)$ if t is a constant
  - $\mu_M(t) = I(f)(\mu_M(t_1), \ldots, \mu_M(t_n))$ if t is $f(t_1, \ldots, t_n)$
- A valuation μ_M for a formula f with respect to a first order structure $M = (\mathcal{D}, I, v)$ is:
  - $\mu_M(\top) = 1$
  - $\mu_M(\bot) = 0$
  - $\mu_M(p(t_1, \ldots, t_n)) = I(p)(\mu_M(t_1), \ldots, \mu_M(t_n))$
  - continuing on next slide

Valuation

- A valuation μ_M of a term t with respect to a first order structure $M = (\mathcal{D}, I, v)$ is:
  - $\mu_M(t) = v(t)$ if t is a variable
  - $\mu_M(t) = I(t)$ if t is a constant
  - $\mu_M(t) = I(f)(\mu_M(t_1), \ldots, \mu_M(t_n))$ if t is $f(t_1, \ldots, t_n)$
- A valuation μ_M for a formula f with respect to a first order structure $M = (\mathcal{D}, I, v)$ is:
  - $\mu_M(\top) = 1$
  - $\mu_M(\bot) = 0$
  - $\mu_M(p(t_1, \ldots, t_n)) = I(p)(\mu_M(t_1), \ldots, \mu_M(t_n))$
  - continuing on next slide

Valuation

A valuation μ_M for a formula f with respect to a first order structure $M = (\mathcal{D}, I, v)$ is:

- $\mu_M(\neg g) = \neg \mu_M(g)$ using propositional logic
- $\mu_M(g \circ h) = \mu_M(g) \circ \mu_M(h)$ using propositional logic for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$
- $\mu_M(\forall V\ g) = 1$ if $\mu_M(g[d/V]) = 1$ for all $d \in \mathcal{D}$
- $\mu_M(\forall V\ g) = 0$ if $\mu_M(g[d/V]) = 0$ for some $d \in \mathcal{D}$
- $\mu_M(\exists V\ g) = 1$ if $\mu_M(g[d/V]) = 1$ for some $d \in \mathcal{D}$
- $\mu_M(\exists V\ g) = 0$ if $\mu_M(g[d/V]) = 0$ for all $d \in \mathcal{D}$

Practical Example 1

$$n(z)$$
$$\forall X\ (n(X) \rightarrow n(s(X)))$$
$$\forall X\ \forall Y\ (\neg(e(X, Y)) \rightarrow \neg(e(s(X), s(Y))))$$
$$\forall X\ \neg e(s(X), z)$$

- $\mathcal{D}$: Natural numbers (including 0)
- $I(z) = 0$
- $I(s)$: Successor of a number
- $I(n)$: True for natural numbers
- $I(e)$: True for equal numbers
- no free variables

Peano

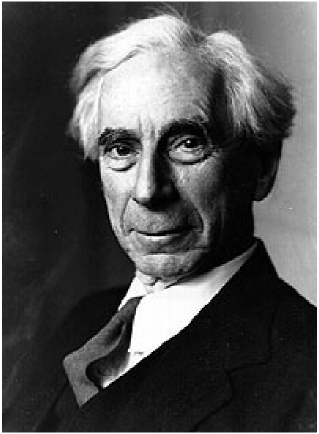
Giuseppe Peano (1858–1932)

Practical Example 2

$$\forall X \neg e(s(X), z)$$
$$\forall X \forall Y (e(s(X), s(Y)) \rightarrow e(X, Y))$$
$$\forall X\ e(a(X, z), X)$$
$$\forall X \forall Y\ e(a(X, s(Y)), s(a(X, Y)))$$
$$\forall X\ e(m(X, z), z)$$
$$\forall X \forall Y\ e(m(X, s(Y)), a(m(X, Y), X))$$

- $\mathcal{D}$: Natural numbers (including 0)
- $I(z) = 0$
- $I(s)$: Successor of a number
- $I(a)$: Sum of two numbers
- $I(m)$: Product of two numbers
- $I(e)$: True for equal numbers
- no free variables

Russell and Whitehead


Bertrand Russell (1872–1970)


Alfred Whitehead (1861–1947)

Socrates Example

$$human(socrates)$$
$$\forall X(human(X) \rightarrow mortal(X))$$
$$mortal(socrates)$$