Logical and Theoretical Foundations of Computer Science

LaTFoCS

Pamela Fleischmann

fpa@informatik.uni-kiel.de

Winter Semester 2019

Kiel University
Dependable Systems Group

Predicate Logic as a Formal Language - Syntax

Necessary Sets and Objects

- set of constants $\mathscr{C}$: Andy, KielUniversity, Room3

Necessary Sets and Objects

- set of constants $\mathscr{C}$: Andy, KielUniversity, Room3
- set of variables $\mathscr{V}$: $x, y, z, x_1, x_2, \dots$

Necessary Sets and Objects

- set of constants $\mathscr{C}$: Andy, KielUniversity, Room3
- set of variables $\mathscr{V}$: $x, y, z, x_1, x_2, \ldots$
- set of function symbols $\mathscr{F}$: $f, g, h, \ldots$ with their arity

Necessary Sets and Objects

- set of constants $\mathscr{C}$: Andy, KielUniversity, Room3
- set of variables $\mathscr{V}$: $x, y, z, x_1, x_2, \ldots$
- set of function symbols $\mathscr{F}$: $f, g, h, \ldots$ with their arity
- set of predicate symbols $\mathscr{P}$: $P, R, S, \ldots$

Necessary Sets and Objects

- set of constants $\mathscr{C}$: Andy, KielUniversity, Room3
- set of variables $\mathscr{V}$: $x, y, z, x_1, x_2, \ldots$
- set of function symbols $\mathscr{F}$: $f, g, h, \ldots$ with their arity
- set of predicate symbols $\mathscr{P}$: $P, R, S, \ldots$

Notice: constants are 0-arity (nullary) functions (no argument, no output)

Terms

Definition

The set of terms $\mathcal{T}$ is inductively defined over $(\mathcal{C}, \mathcal{V}, \mathcal{F})$ by

1. each $x \in \mathcal{V}$ is a term

Terms

Definition

The set of terms $\mathcal{T}$ is inductively defined over $(\mathcal{C}, \mathcal{V}, \mathcal{F})$ by

1. each $x \in \mathcal{V}$ is a term
2. each $c \in \mathcal{C}$ is a term

Terms

Definition

The set of terms $\mathcal{T}$ is inductively defined over $(\mathcal{C}, \mathcal{V}, \mathcal{F})$ by

1. each $x \in \mathcal{V}$ is a term
2. each $c \in \mathcal{C}$ is a term
3. for terms $t_1, \ldots, t_n$, $n \in \mathbb{N}$, a function symbol $f \in \mathcal{F}$ with arity n

$$f(t_1, \ldots, t_n) \text{ is a term}$$

Terms

Definition

The set of terms $\mathcal{T}$ is inductively defined over $(\mathcal{C}, \mathcal{V}, \mathcal{F})$ by

1. each $x \in \mathcal{V}$ is a term
2. each $c \in \mathcal{C}$ is a term
3. for terms $t_1, \ldots, t_n$, $n \in \mathbb{N}$, a function symbol $f \in \mathcal{F}$ with arity n

$$f(t_1, \ldots, t_n) \text{ is a term}$$

4. nothing else is a term

Examples for Terms

- Andy

Examples for Terms

- Andy
- y

Examples for Terms

- Andy
- y
- owner(car with plate KI KI 1234)

Examples for Terms

- Andy
- y
- owner(car with plate KI KI 1234)
- grade(Cindy, LiCS)

Examples for Terms

- Andy
- y
- owner(car with plate KI KI 1234)
- grade(Cindy, LiCS)
- $\mathrm{sum}(1, x, 2, y)$ if sum is 4-ary

Formulae

Definition

The set of first order predicate logic formulae Φ_{FO} is inductively defined over $(\mathcal{T}, \mathcal{F})$ by

1. for terms $t_1, \ldots, t_n \in \mathcal{T}$ and an n-ary predicate symbol $P \in \mathcal{P}$, $n \in \mathbb{N}$

$$P(t_1, \ldots, t_n) \in \Phi_{\mathrm{FO}} \text{ is an atomic formula}$$

Formulae

Definition

The set of first order predicate logic formulae Φ_{FO} is inductively defined over $(\mathcal{T}, \mathcal{F})$ by

1. for terms $t_1, \ldots, t_n \in \mathcal{T}$ and an n-ary predicate symbol $P \in \mathcal{P}$, $n \in \mathbb{N}$

$$P(t_1, \ldots, t_n) \in \Phi_{\mathrm{FO}} \text{ is an atomic formula}$$

2. if $\varphi \in \Phi_{\mathrm{FO}}$ then $\neg\varphi \in \Phi_{\mathrm{FO}}$

Formulae

Definition

The set of first order predicate logic formulae Φ_{FO} is inductively defined over $(\mathcal{T}, \mathcal{F})$ by

1. for terms $t_1, \ldots, t_n \in \mathcal{T}$ and an n-ary predicate symbol $P \in \mathcal{P}$, $n \in \mathbb{N}$

$$P(t_1, \ldots, t_n) \in \Phi_{\mathrm{FO}} \text{ is an atomic formula}$$

2. if $\varphi \in \Phi_{\mathrm{FO}}$ then $\neg\varphi \in \Phi_{\mathrm{FO}}$
3. if $\varphi, \psi \in \Phi_{\mathrm{FO}}$ then $(\varphi \wedge \psi), (\varphi \vee \psi), (\varphi \rightarrow \psi) \in \Phi_{\mathrm{FO}}$

Formulae

Definition

The set of first order predicate logic formulae Φ_{FO} is inductively defined over $(\mathcal{T}, \mathcal{F})$ by

1. for terms $t_1, \ldots, t_n \in \mathcal{T}$ and an n-ary predicate symbol $P \in \mathcal{P}, n \in \mathbb{N}$

$$P(t_1, \ldots, t_n) \in \Phi_{\mathrm{FO}} \text{ is an atomic formula}$$

2. if $\varphi \in \Phi_{\mathrm{FO}}$ then $\neg\varphi \in \Phi_{\mathrm{FO}}$
3. if $\varphi, \psi \in \Phi_{\mathrm{FO}}$ then $(\varphi \wedge \psi), (\varphi \vee \psi), (\varphi \rightarrow \psi) \in \Phi_{\mathrm{FO}}$
4. if $\varphi \in \Phi_{\mathrm{FO}}$ and $x \in \mathcal{V}$ then $(\forall x\varphi), (\exists x\varphi) \in \Phi_{\mathrm{FO}}$

Formulae

Definition

The set of first order predicate logic formulae Φ_{FO} is inductively defined over $(\mathcal{T}, \mathcal{F})$ by

1. for terms $t_1, \ldots, t_n \in \mathcal{T}$ and an n-ary predicate symbol $P \in \mathcal{P}, n \in \mathbb{N}$

$$P(t_1, \ldots, t_n) \in \Phi_{\mathrm{FO}} \text{ is an atomic formula}$$

2. if $\varphi \in \Phi_{\mathrm{FO}}$ then $\neg\varphi \in \Phi_{\mathrm{FO}}$
3. if $\varphi, \psi \in \Phi_{\mathrm{FO}}$ then $(\varphi \wedge \psi), (\varphi \vee \psi), (\varphi \rightarrow \psi) \in \Phi_{\mathrm{FO}}$
4. if $\varphi \in \Phi_{\mathrm{FO}}$ and $x \in \mathcal{V}$ then $(\forall x \varphi), (\exists x \varphi) \in \Phi_{\mathrm{FO}}$
5. nothing else is a formula

Binding Priorities

- $\neg$, $\forall$, $\exists$ bind most tightly
- $\wedge$, $\vee$ are next in the hierarchy
- $\rightarrow$ has loosest connectivity and is right-associative

Example

We want to express:

Every sister of my mother is my aunt.

Example

We want to express:

Every sister of my mother is my aunt.

- constants: I for the person

Example

We want to express:

Every sister of my mother is my aunt.

- constants: I for the person
- functions: mother(x) (mother of x)

Example

We want to express:

Every sister of my mother is my aunt.

- constants: I for the person
- functions: mother(x) (mother of x)
- relations: Sister(x, y), Aunt(x, y) (x is sister/aunt of y)

Example

We want to express:

Every sister of my mother is my aunt.

- constants: I for the person
- functions: mother(x) (mother of x)
- relations: Sister(x, y), Aunt(x, y) (x is sister/aunt of y)

$$(\forall z\, \mathrm{Sister}(z, \mathrm{mother}(I)) \to \mathrm{Aunt}(z, I))$$

Example

We want to express:

Every sister of my mother is my aunt.

- constants: I for the person
- functions: mother(x) (mother of x)
- relations: Sister(x, y), Aunt(x, y) (x is sister/aunt of y)

$$(\forall z\, \mathrm{Sister}(z, \mathrm{mother}(I)) \longrightarrow \mathrm{Aunt}(z, I))$$

Is my mother now my aunt?

Parse Tree of Predicate Logic Formulae

Definition

The parse tree of $\varphi \in \Phi_{\text{FO}}$ is build according to the rules for propositional logic formulae with three additional rules

- $\varphi = (\forall x \psi(x))$: $\forall x$ is a node with child $\psi(x)$
- $\varphi = (\exists x \psi(x))$: $\exists x$ is a node with child $\psi(x)$
- $\varphi = P(t_1, \ldots, t_n)$: P is a node with children $t_1, \ldots, t_n$

Free and bound variables - Scopes

Definition

$\varphi \in \Phi_{FO}$

- x in φ free iff x is leaf in the parse tree and on the path to the root there is no occurrence of $\forall x$ or $\exists x$

Free and bound variables - Scopes

Definition

$\varphi \in \Phi_{\text{FO}}$

- x in φ free iff x is leaf in the parse tree and on the path to the root there is no occurrence of $\forall x$ or $\exists x$
- x in φ bound iff x is a leaf in the parse tree and x is not free

Free and bound variables - Scopes

Definition

$\varphi \in \Phi_{\mathrm{FO}}$

- x in φ free iff x is leaf in the parse tree and on the path to the root there is no occurrence of $\forall x$ or $\exists x$
- x in φ bound iff x is a leaf in the parse tree and x is not free
- if φ does not contain free variables it is called closed

Free and bound variables - Scopes

Definition

$\varphi \in \Phi_{\mathrm{FO}}$

- x in φ free iff x is leaf in the parse tree and on the path to the root there is no occurrence of $\forall x$ or $\exists x$
- x in φ bound iff x is a leaf in the parse tree and x is not free
- if φ does not contain free variables it is called closed

Definition

A variable $x \in \mathcal{V}$ is bound

Free and bound variables - Scopes

Definition

$\varphi \in \Phi_{FO}$

- x in φ free iff x is leaf in the parse tree and on the path to the root there is no occurrence of $\forall x$ or $\exists x$
- x in φ bound iff x is a leaf in the parse tree and x is not free
- if φ does not contain free variables it is called closed

Definition

A variable $x \in \mathcal{V}$ is bound

- universally if x occurs as $(\forall x \psi)$

Free and bound variables - Scopes

Definition

$\varphi \in \Phi_{FO}$

- x in φ free iff x is leaf in the parse tree and on the path to the root there is no occurrence of $\forall x$ or $\exists x$
- x in φ bound iff x is a leaf in the parse tree and x is not free
- if φ does not contain free variables it is called closed

Definition

A variable $x \in \mathcal{V}$ is bound

- universally if x occurs as $(\forall x \psi)$
- existentially if x occurs as $(\exists x \psi)$

Free and bound variables - Scopes

Definition

$\varphi \in \Phi_{\text{FO}}$

- x in φ free iff x is leaf in the parse tree and on the path to the root there is no occurrence of $\forall x$ or $\exists x$
- x in φ bound iff x is a leaf in the parse tree and x is not free
- if φ does not contain free variables it is called closed

Definition

A variable $x \in \mathscr{V}$ is bound

- universally if x occurs as $(\forall x \psi)$
- existentially if x occurs as $(\exists x \psi)$

ψ is called the scope of x

Universal and Existential Closure

Definition

$\varphi \in \Phi_{\text{FO}}$

- The set of all free variables of φ is denoted by $\text{Free}(\varphi)$.

Universal and Existential Closure

Definition

$\varphi \in \Phi_{\mathrm{FO}}$

- The set of all free variables of φ is denoted by $\mathrm{Free}(\varphi)$.
- If $\{x_1, \ldots, x_n\} = \mathrm{Free}(\varphi)$ then $\forall x_1 \ldots \forall x_n(\varphi)$ is the universal closure of φ.

Universal and Existential Closure

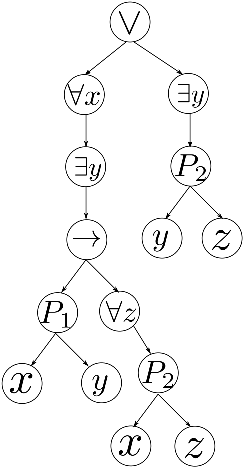
Definition

$\varphi \in \Phi_{\mathrm{FO}}$

- The set of all free variables of φ is denoted by $\mathrm{Free}(\varphi)$.
- If $\{x_1, \ldots, x_n\} = \mathrm{Free}(\varphi)$ then $\forall x_1 \ldots \forall x_n(\varphi)$ is the universal closure of φ.
- If $\{x_1, \ldots, x_n\} = \mathrm{Free}(\varphi)$ then $\exists x_1 \ldots \exists x_n(\varphi)$ is the existential closure of φ.

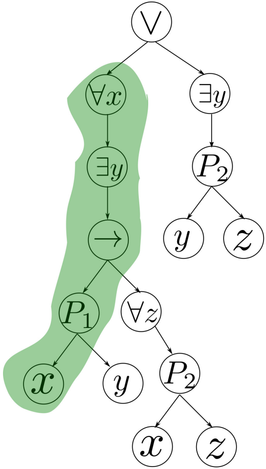
Example

$$(\forall x(\exists y P_1(x,y) \rightarrow \forall z P_2(x,z)) \vee (\exists y P_2(y,z)))$$



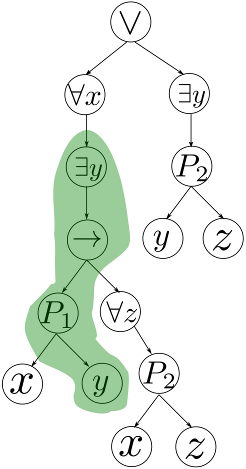
Example

$$(\forall x(\exists y P_1(x,y) \rightarrow \forall z P_2(x,z)) \vee (\exists y P_2(y,z)))$$



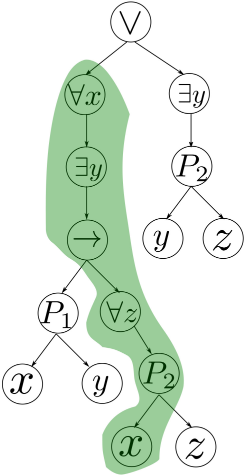
Example

$$(\forall x(\exists y P_1(x,y) \rightarrow \forall z P_2(x,z)) \vee (\exists y P_2(y,z)))$$



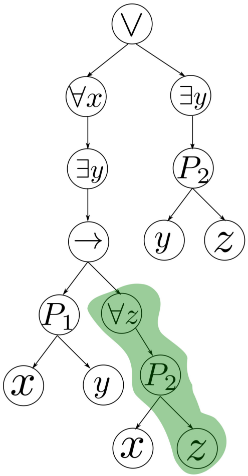
Example

$$(\forall x(\exists y P_1(x,y) \rightarrow \forall z P_2(x,z)) \vee (\exists y P_2(y,z)))$$



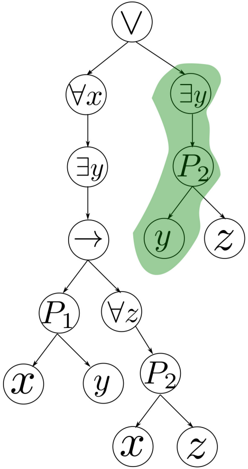
Example

$$(\forall x(\exists y P_1(x,y) \rightarrow \forall z P_2(x,z)) \vee (\exists y P_2(y,z)))$$



Example

$$(\forall x(\exists y P_1(x,y) \longrightarrow \forall z P_2(x,z)) \vee (\exists y P_2(y,z)))$$



Example

$$(\forall x(\exists y P_1(x,y) \longrightarrow \forall z P_2(x,z)) \vee (\exists y P_2(y,z)))$$