# Lógica de predicados

Matemática estructural y lógica

ISIS-1104

# ¿Por qué las proposiciones no bastan?

# ¿Por qué las proposiciones no bastan?

Si tenemos el siguiente enunciado

# ¿Por qué las proposiciones no bastan?

Si tenemos el siguiente enunciado

*Toda la comida italiana es deliciosa.*
*La pizza es comida italiana*
*Luego la pizza es deliciosa*

# ¿Por qué las proposiciones no bastan?

Si tenemos el siguiente enunciado

*Toda la comida italiana es deliciosa.*
*La pizza es comida italiana*
*Luego la pizza es deliciosa*

La lógica proposicional es insuficiente para expresar casos particulares y generalizaciones.

# Predicados

# Predicados

Un predicado sobre $T$ es una función $T \to Bool$.

# Predicados

Un predicado sobre $T$ es una función $T \to Bool$.

$T$ puede ser cualquier conjunto (tipo) de elementos. Por ejemplo:

# Predicados

Un predicado sobre $T$ es una función $T \to Bool$.

$T$ puede ser cualquier conjunto (tipo) de elementos. Por ejemplo:

- Si tenemos $k : Int$, entonces

  $$P(k) = k > 0$$

  es un predicado sobre $Int$.

# Predicados

Un predicado sobre $T$ es una función $T \to Bool$.

$T$ puede ser cualquier conjunto (tipo) de elementos. Por ejemplo:

- Si tenemos $k : Int$, entonces

  $$P(k) = k > 0$$

  es un predicado sobre $Int$.
- Si tenemos $s : Persona$, entonces

  $$esAdulto(s) = s \text{ es un adulto}$$

  es un predicado sobre $Persona$.

# Predicados

Un predicado sobre $T$ es una función $T \to Bool$.

$T$ puede ser cualquier conjunto (tipo) de elementos. Por ejemplo:

- Si tenemos $k : Int$, entonces

  $$P(k) = k > 0$$

  es un predicado sobre $Int$.
- Si tenemos $s : Persona$, entonces

  $$esAdulto(s) = s \text{ es un adulto}$$

  es un predicado sobre $Persona$.

Como son funciones, los predicados no tienen un valor de verdad definido hasta que se evalúan en algún $t : T$.

# Predicados

## Predicados

Los predicados pueden combinarse usando las operaciones lógicas que ya conocemos.

# Predicados

Los predicados pueden combinarse usando las operaciones lógicas que ya conocemos.

Por ejemplo si tenemos los predicados

# Predicados

Los predicados pueden combinarse usando las operaciones lógicas que ya conocemos.

Por ejemplo si tenemos los predicados

$$esAdulto(k) = k \text{ es un adulto}$$

# Predicados

Los predicados pueden combinarse usando las operaciones lógicas que ya conocemos.

Por ejemplo si tenemos los predicados

$$esAdulto(k) = k \text{ es un adulto}$$

$$sabeConducir(k) = k \text{ sabe conducir}$$

# Predicados

Los predicados pueden combinarse usando las operaciones lógicas que ya conocemos.

Por ejemplo si tenemos los predicados

$$esAdulto(k) = k \text{ es un adulto}$$

$$sabeConducir(k) = k \text{ sabe conducir}$$

Podemos definir un nuevo predicado

# Predicados

Los predicados pueden combinarse usando las operaciones lógicas que ya conocemos.

Por ejemplo si tenemos los predicados

$$esAdulto(k) = k \text{ es un adulto}$$

$$sabeConducir(k) = k \text{ sabe conducir}$$

Podemos definir un nuevo predicado

$$esAdulto(k) \wedge \neg sabeConducir(k)$$

# Predicados

# Predicados

Todos los axiomas y reglas de inferencia de lógica proposicional siguen siendo válidos.

# Predicados

Todos los axiomas y reglas de inferencia de lógica proposicional siguen siendo válidos.

Por ejemplo

# Predicados

Todos los axiomas y reglas de inferencia de lógica proposicional siguen siendo válidos.

Por ejemplo

- $P(x) \wedge P(x) \equiv P(x)$

# Predicados

Todos los axiomas y reglas de inferencia de lógica proposicional siguen siendo válidos.

Por ejemplo

- $P(x) \wedge P(x) \equiv P(x)$
- $P(x) \Rightarrow Q(y) \equiv \neg P(x) \vee Q(y)$

# Predicados

Todos los axiomas y reglas de inferencia de lógica proposicional siguen siendo válidos.

Por ejemplo

- $P(x) \land P(x) \equiv P(x)$
- $P(x) \Rightarrow Q(y) \equiv \neg P(x) \lor Q(y)$
- $\neg(P(x) \lor Q(x)) \equiv \neg P(x) \land \neg Q(x)$

# Predicados

Todos los axiomas y reglas de inferencia de lógica proposicional siguen siendo válidos.

Por ejemplo

- $P(x) \wedge P(x) \equiv P(x)$
- $P(x) \Rightarrow Q(y) \equiv \neg P(x) \vee Q(y)$
- $\neg(P(x) \vee Q(x)) \equiv \neg P(x) \wedge \neg Q(x)$
- $P(x) \wedge (P(x) \Rightarrow Q(x)) \Rightarrow Q(x)$

# Predicados

Todos los axiomas y reglas de inferencia de lógica proposicional siguen siendo válidos.

Por ejemplo

- $P(x) \wedge P(x) \equiv P(x)$
- $P(x) \Rightarrow Q(y) \equiv \neg P(x) \vee Q(y)$
- $\neg(P(x) \vee Q(x)) \equiv \neg P(x) \wedge \neg Q(x)$
- $P(x) \wedge (P(x) \Rightarrow Q(x)) \Rightarrow Q(x)$

Sin embargo hay que ser cuidadosos con las variables de los predicados

# Predicados

Todos los axiomas y reglas de inferencia de lógica proposicional siguen siendo válidos.

Por ejemplo

- $P(x) \wedge P(x) \equiv P(x)$
- $P(x) \Rightarrow Q(y) \equiv \neg P(x) \vee Q(y)$
- $\neg(P(x) \vee Q(x)) \equiv \neg P(x) \wedge \neg Q(x)$
- $P(x) \wedge (P(x) \Rightarrow Q(x)) \Rightarrow Q(x)$

Sin embargo hay que ser cuidadosos con las variables de los predicados

$$P(x) \Rightarrow Q(y) \not\equiv P(y) \Rightarrow Q(x)$$

# Constantes

# Constantes

En ocasiones, queremos hacer referencia a algún elemento de $T$, para esto usamos constantes.

# Constantes

En ocasiones, queremos hacer referencia a algún elemento de $T$, para esto usamos constantes.

Por ejemplo si tenemos el tipo *Animal* y el predicado

# Constantes

En ocasiones, queremos hacer referencia a algún elemento de $T$, para esto usamos constantes.

Por ejemplo si tenemos el tipo *Animal* y el predicado

$$esGato(k) = k \text{ es un gato}$$

# Constantes

En ocasiones, queremos hacer referencia a algún elemento de $T$, para esto usamos constantes.

Por ejemplo si tenemos el tipo *Animal* y el predicado

$$esGato(k) = k \text{ es un gato}$$

podemos definir una constante $garfield : Animal$ y escribir

# Constantes

En ocasiones, queremos hacer referencia a algún elemento de $T$, para esto usamos constantes.

Por ejemplo si tenemos el tipo *Animal* y el predicado

$$esGato(k) = k \text{ es un gato}$$

podemos definir una constante $garfield : Animal$ y escribir

$$esGato(garfield)$$

## Constantes

En ocasiones, queremos hacer referencia a algún elemento de $T$, para esto usamos constantes.

Por ejemplo si tenemos el tipo *Animal* y el predicado

$$esGato(k) = k \text{ es un gato}$$

podemos definir una constante $garfield : Animal$ y escribir

$$esGato(garfield)$$

para afirmar que la constante *garfield* cumple el predicado *esGato*

# Cuantificador universal

# Cuantificador universal

Si tenemos el cuantificador

$$(\wedge x : T \mid P(x) : Q(x))$$

# Cuantificador universal

Si tenemos el cuantificador

$$(\wedge x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

# Cuantificador universal

Si tenemos el cuantificador

$$(\wedge x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

Para todos los $x$ tales que $P(x)$, se cumple $Q(x)$.

# Cuantificador universal

Si tenemos el cuantificador

$$(\wedge x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

Para todos los $x$ tales que $P(x)$, se cumple $Q(x)$.

Usualmente cambiamos el símbolo $\wedge$

$$(\forall x : T \mid P(x) : Q(x))$$

# Cuantificador universal

Si tenemos el cuantificador

$$(\wedge x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

Para todos los $x$ tales que $P(x)$, se cumple $Q(x)$.

Usualmente cambiamos el símbolo $\wedge$

$$(\forall x : T \mid P(x) : Q(x))$$

Por ejemplo

# Cuantificador universal

Si tenemos el cuantificador

$$(\wedge x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

Para todos los $x$ tales que $P(x)$, se cumple $Q(x)$.

Usualmente cambiamos el símbolo $\wedge$

$$(\forall x : T \mid P(x) : Q(x))$$

Por ejemplo

$$(\forall x : \mathit{Animal} \mid \mathit{esGato}(x) : \mathit{tieneBigotes}(x))$$

# Cuantificador universal

Si tenemos el cuantificador

$$(\wedge x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

Para todos los $x$ tales que $P(x)$, se cumple $Q(x)$.

Usualmente cambiamos el símbolo $\wedge$

$$(\forall x : T \mid P(x) : Q(x))$$

Por ejemplo

$$(\forall x : Animal \mid esGato(x) : tieneBigotes(x))$$

$$(\forall x : Animal \mid esGato(x) : sabeVolar(x))$$

# Cuantificador existencial

# Cuantificador existencial

Si tenemos el cuantificador

$$(\vee x : T \mid P(x) : Q(x))$$

# Cuantificador existencial

Si tenemos el cuantificador

$$(\vee x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

# Cuantificador existencial

Si tenemos el cuantificador

$$(\vee x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

Existen $x$ tales que $P(x)$, para los cuales se cumple $Q(x)$.

# Cuantificador existencial

Si tenemos el cuantificador

$$(\vee x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

> Existen $x$ tales que $P(x)$, para los cuales se cumple $Q(x)$.

Usualmente cambiamos el símbolo $\wedge$

$$(\exists x : T \mid P(x) : Q(x))$$

## Cuantificador existencial

Si tenemos el cuantificador

$$(\vee x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

> Existen $x$ tales que $P(x)$, para los cuales se cumple $Q(x)$.

Usualmente cambiamos el símbolo $\wedge$

$$(\exists x : T \mid P(x) : Q(x))$$

Por ejemplo

# Cuantificador existencial

Si tenemos el cuantificador

$$(\vee x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

Existen $x$ tales que $P(x)$, para los cuales se cumple $Q(x)$.

Usualmente cambiamos el símbolo $\wedge$

$$(\exists x : T \mid P(x) : Q(x))$$

Por ejemplo

$$(\exists x : Animal \mid esGato(x) : comeLasagna(x))$$

# Cuantificador existencial

Si tenemos el cuantificador

$$(\vee x : T \mid P(x) : Q(x))$$

Con $P$ y $Q$ predicados sobre $T$, podemos darle la interpretación

Existen $x$ tales que $P(x)$, para los cuales se cumple $Q(x)$.

Usualmente cambiamos el símbolo $\wedge$

$$(\exists x : T \mid P(x) : Q(x))$$

Por ejemplo

$$(\exists x : Animal \mid esGato(x) : comeLasagna(x))$$

$$(\exists x : Animal \mid esGato(x) : esPerro(x))$$

# De vuelta a la pizza

# De vuelta a la pizza

Si tomamos $C$ como el tipo de la comida. Y $pizza : C$ como una constante.

# De vuelta a la pizza

Si tomamos $C$ como el tipo de la comida. Y $pizza : C$ como una constante.

*Toda la comida italiana es deliciosa.*
*La pizza es comida italiana.*
*Luego la pizza es deliciosa.*

# De vuelta a la pizza

Si tomamos $C$ como el tipo de la comida. Y $pizza : C$ como una constante.

*Toda la comida italiana es deliciosa.*
*La pizza es comida italiana.*
*Luego la pizza es deliciosa.*

Puede escribirse como

$$\frac{(\forall c : C \mid esItaliana(c) : esDeliciosa(c)),\ esItaliana(pizza)}{esDeliciosa(pizza)}$$

## Variables acotadas

## Variables acotadas

Es necesario distinguir que variables están libres y cuales están acotadas en éste nuevo lenguaje.

# Variables acotadas

Es necesario distinguir que variables están libres y cuales están acotadas en éste nuevo lenguaje.

Por ejemplo:

## Variables acotadas

Es necesario distinguir que variables están libres y cuales están acotadas en éste nuevo lenguaje.

Por ejemplo:

- $x$ es libre en $x > 5$.

# Variables acotadas

Es necesario distinguir que variables están libres y cuales están acotadas en éste nuevo lenguaje.

Por ejemplo:

- $x$ es libre en $x > 5$.
- $x$ es libre en $esGato(x)$.

# Variables acotadas

Es necesario distinguir que variables están libres y cuales están acotadas en éste nuevo lenguaje.

Por ejemplo:

- $x$ es libre en $x > 5$.
- $x$ es libre en $esGato(x)$.
- $n$ está acotada en $(\exists n : Int \mid esPar(n) : n > 0)$.

# Variables acotadas

Es necesario distinguir que variables están libres y cuales están acotadas en éste nuevo lenguaje.

Por ejemplo:

- $x$ es libre en $x > 5$.
- $x$ es libre en $esGato(x)$.
- $n$ está acotada en $(\exists n : Int \mid esPar(n) : n > 0)$.
- $k$ es libre en $(\forall n : Int \mid esPrimo(n) : n < k)$.

# Variables acotadas

Es necesario distinguir que variables están libres y cuales están acotadas en éste nuevo lenguaje.

Por ejemplo:

- $x$ es libre en $x > 5$.
- $x$ es libre en $esGato(x)$.
- $n$ está acotada en $(\exists n : Int \mid esPar(n) : n > 0)$.
- $k$ es libre en $(\forall n : Int \mid esPrimo(n) : n < k)$.

Entonces

# Variables acotadas

Es necesario distinguir que variables están libres y cuales están acotadas en éste nuevo lenguaje.

Por ejemplo:

- $x$ es libre en $x > 5$.
- $x$ es libre en $esGato(x)$.
- $n$ está acotada en $(\exists n : Int \mid esPar(n) : n > 0)$.
- $k$ es libre en $(\forall n : Int \mid esPrimo(n) : n < k)$.

Entonces

- $x$ está ligada si es la variable sobre la que estoy cuantificando.

# Variables acotadas

Es necesario distinguir que variables están libres y cuales están acotadas en éste nuevo lenguaje.

Por ejemplo:

- $x$ es libre en $x > 5$.
- $x$ es libre en $esGato(x)$.
- $n$ está acotada en $(\exists n : Int \mid esPar(n) : n > 0)$.
- $k$ es libre en $(\forall n : Int \mid esPrimo(n) : n < k)$.

Entonces

- $x$ está ligada si es la variable sobre la que estoy cuantificando.
- $x$ es libre si no estoy cuantificando sobre ella.

# Variables acotadas

# Variables acotadas

Si $x$ es libre en una cuantificación, decimos que la cuantificación depende de $x$.

## Variables acotadas

Si $x$ es libre en una cuantificación, decimos que la cuantificación depende de $x$.

- $esGato(a)$ depende de $a$.

# Variables acotadas

Si $x$ es libre en una cuantificación, decimos que la cuantificación depende de $x$.

- $esGato(a)$ depende de $a$.
- $(\exists n : Int \mid: n > k)$ depende de $k$.

## Variables acotadas

Si $x$ es libre en una cuantificación, decimos que la cuantificación depende de $x$.

- $\mathit{esGato}(a)$ depende de $a$.
- $(\exists n : \mathit{Int} \mid: n > k)$ depende de $k$.
- $(\exists n : \mathit{Int} \mid: n > 0)$ no depende de $n$.

# Variables acotadas

Si $x$ es libre en una cuantificación, decimos que la cuantificación depende de $x$.

- $\mathit{esGato}(a)$ depende de $a$.
- $(\exists n : \mathit{Int} \mid : n > k)$ depende de $k$.
- $(\exists n : \mathit{Int} \mid : n > 0)$ no depende de $n$.

Cuando una cuantificación depende de alguna variable podemos usarla como una proposición

# Variables acotadas

Si $x$ es libre en una cuantificación, decimos que la cuantificación depende de $x$.

- $esGato(a)$ depende de $a$.
- $(\exists n : Int \,|\!: n > k)$ depende de $k$.
- $(\exists n : Int \,|\!: n > 0)$ no depende de $n$.

Cuando una cuantificación depende de alguna variable podemos usarla como una proposición

$$Q(k) = (\exists n : Int \,|\!: k = 2n)$$

# Variables acotadas

Si $x$ es libre en una cuantificación, decimos que la cuantificación depende de $x$.

- $esGato(a)$ depende de $a$.
- $(\exists n : Int \mid : n > k)$ depende de $k$.
- $(\exists n : Int \mid : n > 0)$ no depende de $n$.

Cuando una cuantificación depende de alguna variable podemos usarla como una proposición

$$Q(k) = (\exists n : Int \mid : k = 2n)$$

entonces

# Variables acotadas

Si $x$ es libre en una cuantificación, decimos que la cuantificación depende de $x$.

- $esGato(a)$ depende de $a$.
- $(\exists n : Int \mid: n > k)$ depende de $k$.
- $(\exists n : Int \mid: n > 0)$ no depende de $n$.

Cuando una cuantificación depende de alguna variable podemos usarla como una proposición

$$Q(k) = (\exists n : Int \mid: k = 2n)$$

entonces

$$(\forall z : Int \mid esPar(z) : Q(z))$$

## Variables acotadas

Si $x$ es libre en una cuantificación, decimos que la cuantificación depende de $x$.

- $esGato(a)$ depende de $a$.
- $(\exists n : Int \mid: n > k)$ depende de $k$.
- $(\exists n : Int \mid: n > 0)$ no depende de $n$.

Cuando una cuantificación depende de alguna variable podemos usarla como una proposición

$$Q(k) = (\exists n : Int \mid: k = 2n)$$

entonces

$$(\forall z : Int \mid esPar(z) : Q(z))$$

es lo mismo que

# Variables acotadas

Si $x$ es libre en una cuantificación, decimos que la cuantificación depende de $x$.

- $esGato(a)$ depende de $a$.
- $(\exists n : Int \mid : n > k)$ depende de $k$.
- $(\exists n : Int \mid : n > 0)$ no depende de $n$.

Cuando una cuantificación depende de alguna variable podemos usarla como una proposición

$$Q(k) = (\exists n : Int \mid : k = 2n)$$

entonces

$$(\forall z : Int \mid esPar(z) : Q(z))$$

es lo mismo que

$$(\forall z : Int \mid esPar(z) : (\exists n : Int \mid : z = 2n))$$

# Ahora ustedes

# Ahora ustedes

¿Cuáles de las siguientes variables son libres?

## Ahora ustedes

¿Cuáles de las siguientes variables son libres?

- $x$ en

$$x + 1 = 5$$

## Ahora ustedes

¿Cuáles de las siguientes variables son libres?

- $x$ en

$$x + 1 = 5$$

- $a$, $b$ en

$$(\exists a : Animal \mid esGato(n) : sonAmigos(a, b))$$

# Ahora ustedes

¿Cuáles de las siguientes variables son libres?

- $x$ en

$$x + 1 = 5$$

- $a$, $b$ en

$$(\exists a : Animal \mid esGato(n) : sonAmigos(a, b))$$

- $x$, $y$, $z$ en

$$(\forall z : Int \mid z > 0 : (\exists y : Int \mid y > 0 : x^2 + y^2 = z^2))$$