

# Lógica de predicados (de primer orden)

Lógica computacional

24 de abril de 2024

Al tratar con objetos matemáticos, como seguramente ya lo han hecho en los semestres que llevan de la carrera, por lo regular se buscan relaciones generales de todos los objetos de un conjunto, o reconocer un elemento muy particular que sólo tiene ese sentido de particularidad cuando se piensa en todos los demás.

En la lógica de proposiciones sólo se hablaba de los átomos, las proposiciones, y las posibles operaciones sobre ellos, pero nada se hablaba de la relación entre átomos. Eso nos ahorra tener que dar enunciados lógicos para cada elemento de un conjunto y poder hablar de una generalidad, o de una singularidad que se diferencia de una generalidad.

De nueva cuenta definimos este lenguaje de manera recursiva, pero antes de ponerlo formalmente hay que listar lo que necesitamos (para ir a la tienda matemática y conseguir nuestra lista del súper):

1. Las *variables*, que trataran de la generalidad, cuando queramos referir a la singularidad usaremos *constantes*, en contraposición.
2. *Símbolos predicados*, los que dará las características a las variables o constantes.
3. Los *cuantificadores*, que justo nos dirán si una propiedad se cumple para todas las variables, o sólo para una o alguna.

Realmente para definir algunos de estos elementos vienen en diversas presentaciones:

**Definición 1** *Un lenguaje de lógica de predicados consiste de los siguientes símbolos fundamentales:*

■ *Símbolos lógicos*

- Variables:  $x, y, z, \dots, x_0, y_0, z_0, \dots, x_i, y_i, z_i, \dots$
- Conectores lógicos:  $\neg, \wedge, \vee, \rightarrow, \iff$
- Coma y paréntesis:  $, ( )$
- Cuantificadores:  $\exists, \forall$

■ *Símbolos específicos*

- Símbolos predicados:  $P, Q, R, \dots, P_0, Q_0, R_0, \dots, P_i, \dots$
- Símbolos de constantes:  $a, b, \dots, a_0, b_0, \dots, a_i, b_i, \dots$
- Símbolos de funciones:  $f, g, f_0, g_0, \dots, f_i, \dots$

Los símbolos predicados tendrán una *aridad* dependiendo de cuantos símbolos sean parte de su argumento. Cada cuantificador es dual al otro, es decir que puede traducirse:

- $(\forall x)Q(x) \iff \neg(\exists x)\neg Q(x)$
- $(\exists x)Q(x) \iff \neg(\forall x)\neg Q(x)$

**Definición 2** Un término se define inductivamente como:

- Una **constante** es un término.
- una **variable** es un término.
- Si  $f$  es una **función** de aridad- $n$ , y  $t_1, \dots, t_n$  son términos,  $f(t_1, \dots, t_n)$  es un término.

**Definición 3** Una **fórmula atómica** o **átomo** es toda secuencia de símbolos  $P(t_1, \dots, t_n)$  donde  $P$  es un símbolo predicado de aridad- $n$  y  $t_i$  son términos, para  $i = 1, 2, \dots, n$ .

**Definición 4** Una **fórmula** es definida inductivamente como:

- Todo **átomo** es una fórmula
- Si  $\sigma_1$  y  $\sigma_2$  son fórmulas,  $\neg\sigma_1$ ,  $\sigma_1 \wedge \sigma_2$ ,  $\sigma_1 \vee \sigma_2$ ,  $\sigma_1 \rightarrow \sigma_2$  y  $\sigma_1 \iff \sigma_2$  son fórmulas también.
- Si  $v$  es una variable y  $\phi$  una fórmula entonces  $((\exists v)\phi)$  y  $((\forall v)\phi)$  también son fórmulas.
- Solo las secuencias de símbolos formadas de acuerdo a los puntos anteriores son fórmulas.

Bueno, ¿y eso para qué sirve o qué tiene que ver con programación? Pues lo prometido es deuda, veamos un poco de Haskell.

Definamos todos los posibles factores de un número natural cualquiera:

```
fact :: Int -> [Int]
fact n = [x | x <- [1..n], n `mod` x == 0]
```

Todos los número natural menores a un número natural dado:

```
men :: Int -> [Int]
men n = [x | x <- [1..n], x < n]
```

Algunos ejemplos. ¿Cómo escribirían las siguientes frases de canciones populares (de hace años)?

- “Todos tienen tortita menos yo”  $\exists x.\neg T(x)$
- “Siempre existe un roto para un descosido”  $\forall d(\exists r.R(d,r))$
- “Si tienes un hondo pesar, piensa en mí”  $\exists.hP(x)$
- “De noche todos los gatos son pardos”  $\forall x(\exists t.Gato(x) \wedge Pardo(x,t))$

## Variables libres y ligadas

**Definición 5** ■ La ocurrencia de una variable  $v$  en una fórmula  $\phi$  se llama **ligada** si existe una subfórmula  $\psi$  de  $\phi$  que contiene tal ocurrencia y empieza con  $\forall v$  o  $\exists v$ .

■ Una ocurrencia de una variable  $v$  en una fórmula se llama **libre** si no está ligada.

La idea intuitiva (esa es una palabra poco usada en este curso) de libre o ligada es que en el primer caso podemos hacer uso de ella a nuestro antojo, en caso de ser ligada no es tan fácil pues hay constricciones que nos evitan hacer lo que nos da gana con ella. Si un cuantificador está asociado a esa variable se darán cuenta de que hay una construcción, esa variable tiene una función por cumplir en la fórmula, y no es que podamos disponer de ella a nuestro gusto, sólo dentro de las limitaciones que nos permite el cuantificador, sea  $\forall$  o  $\exists$ .

En general hablamos de una **variable libre** en la fórmula  $\phi$  si tiene al menos una libre ocurrencia en  $\phi$ . Por el contrario la variable es ligada si no es libre.

Por ejemplo, si tenemos la fórmula:

$$\phi' : \forall x.P(x, y)$$

Vemos que  $y$  es una variable libre, y  $x$  es una variable ligada. Pero de forma más general, si tenemos la fórmula:

$$\begin{aligned}\phi' : \forall x.(\exists y.(P(x) \vee Q(x, y)) \rightarrow \neg R(x)) \text{ donde podemos subdividir} \\ \psi' : \exists y.(P(x) \vee Q(x, y)) \rightarrow \neg R(x)\end{aligned}$$

Para la subfórmula  $\psi'$ ,  $x$  tiene una ocurrencia libre, pero en la fórmula completa  $\phi'$  está ligada. Observamos que podemos hablar de libre localmente, pero no implica libre globalmente.

**Definición 6** Un término sin variables es llamado un término base.

**Definición 7** Una **fórmula cerrada** es una fórmula sin variables libres.

## Sustituciones

**Definición 8** Un conjunto de sustitución, o simplemente una sustitución, es el conjunto:

$$\theta = \{x_1/t_1, x_2/t_2, \dots, x_n/t_n\},$$

donde  $x_i$  y  $t_i$ ,  $1 \leq i \leq n$ , son correspondientemente variables y términos tales que si  $x_i = x_j$ , entonces  $t_i = t_j$ ,  $i \leq j \leq n$ . Otra forma de escribirlo es:

$$\alpha\theta = \alpha_{x_1:=t_1, x_2:=t_2, \dots, x_n:=t_n}$$

donde  $\alpha$  es un término (fórmula, variable, constante, predicado o la combinación posible de ellos).

Para expresar la sustitución, por lo regular se tiene la expresión (átomo, término o fórmula)  $\phi$  y la sustitución  $\theta$ . La expresión con la sustitución se escribe  $\phi\theta$ . La sustitución vacía se expresa como  $E$ ,  $E = \{\}$ .

**Ejemplo:** Sea la fórmula

$$\phi(x, y, z) : \exists y.R(x, y) \wedge \exists z.\neg Q(x, z),$$

y el término base  $f(a, b)$ . Aplica la sustitución  $[x/f(a, b)]$  en  $\phi(x, y, z)$ .

**Solución:** Es un conjunto pequeño, sólo es una sustitución:

$$\phi(f(a, b), y, z) : \exists y.R(f(a, b), y) \wedge \exists z.\neg Q(f(a, b), z)$$

Ya sabemos que se puede sustituir. pero si definimos la sustitución como un conjunto ¿porque no operar ahora sobre sustituciones? Esa operación básica será la composición, y la composición de dos o más sustituciones es una sustitución, el neutro de esa operación es la sustitución vacía y también existe un inverso.

**Definición 9** Sean  $\theta = \{u_1/s_1, \dots, u_m/s_m\}$  y  $\psi = \{v_1/t_1, \dots, v_n/t_n\}$  dos sustituciones, la composición de ambas es la sustitución:

$$\begin{aligned} \theta\psi = & \{u_1/s_1\psi, \dots, u_m/s_m\psi, v_1/t_1, \dots, v_n/t_n\} \\ & - (\{u_i/s_i\psi \mid u_i = s_i\psi\} \cup \{v_i/t_i \mid v_i \in \{u_1, \dots, u_m\}\}). \end{aligned}$$

Es decir, aplicamos la sustitución  $\psi$  a  $\theta$  y rellenamos con los elementos de  $\psi$ , pero omitimos los términos  $u_i$  de  $\theta$  que toman la forma  $s_i\psi$ , y los términos  $v_i$  de  $\psi$  que contiene las variables  $u_j$  de  $\theta$ .

**Ejemplo:** Sea  $\theta = \{x/f(x), y/z\}$  (lo que sería  $\{u_i/s_i\}$ ) y  $\psi = \{x/a, y/b, z/y\}$  (lo que sería  $\{v_i/t_i\}$ ) dos sustituciones. ¿Cómo sería la sustitución de ambas?

**Respuesta:**

$$\begin{aligned} \theta\psi = & \{x/f(y)\psi, y/z\psi, x/a, y/b, z/y\} \\ & - (\{u_i/s_i\psi \mid u_i = s_i\psi\} \cup \{v_i/t_i \mid v_i \in \{u_1, \dots, u_m\}\}) \\ = & \{x/f(y)\psi, y/z\psi, x/a, y/b, z/y\} - (\{y/y\} \cup \{x/a, y/b\}) \\ = & \{x/f(b), z/y\} \end{aligned}$$

Algunas reglas para poder hacer las sustituciones que son útiles (en la ora forma de escribirlo):

- $x_{[x:=t]} = t$
- $y_{[x:=t]} = y$  siempre que  $y \neq x$
- $c_{[x:=t]} = c$  si  $c \in C$ , es una constante
- $f_m^n(t_1, \dots, t_n)_{[x:=t]} = f_m^n(t_{1[x:=t]}, \dots, t_{n[x:=t]})$
- $P_m^n(t_1, \dots, t_n)_{[x:=t]} = P_m^n(t_{1[x:=t]}, \dots, t_{n[x:=t]})$
- $\neg(\alpha)_{[x:=t]} = \neg(\alpha_{[x:=t]})$
- $(\alpha \vee \beta)_{[x:=t]} = (\alpha_{[x:=t]} \vee \beta_{[x:=t]})$

- $(\alpha \wedge \beta)_{[x:=t]} = (\alpha_{[x:=t]} \wedge \beta_{[x:=t]})$
- $(\alpha \implies \beta)_{[x:=t]} = (\alpha_{[x:=t]} \implies \beta_{[x:=t]})$
- $(\alpha \iff \beta)_{[x:=t]} = (\alpha_{[x:=t]} \iff \beta_{[x:=t]})$
- $(\forall x.\alpha)_{[x:=t]} = \forall x.\alpha$
- $(\forall y.\alpha)_{[x:=t]} = \forall x.(\alpha)_{[x:=t]}$  si  $x \notin F(\alpha)$  ni  $y \notin t$ .
- $(\forall y.\alpha)_{[x:=t]} = \forall z.(\alpha_{[y:=z]})_{[x:=t]}$  si  $x \in F(\alpha)$  ni  $y \in t$ .
- $(\exists x.\alpha)_{[x:=t]} = \exists x.\alpha$
- $(\exists y.\alpha)_{[x:=t]} = \exists x.(\alpha)_{[x:=t]}$  si  $x \notin F(\alpha)$  ni  $y \notin t$ .
- $(\exists y.\alpha)_{[x:=t]} = \exists z.(\alpha_{[y:=z]})_{[x:=t]}$  si  $x \in F(\alpha)$  y  $y \in t$ .

Para la conjunción, disyunción, implicación y equivalencia me parece que la cuestión es clara (pero puedo equivocarme, si no les queda claro no dejen de preguntar), pero los casos de los cuantificadores pueden necesitar una revisión extra.

Si la variable a cambiar está ligada, no se cambia, como se definió en su caso, si la variable está ligada nos impide operar sobre ella de esa forma, no tenemos libertad sobre ella. Si no está ligada podemos cambiarla con toda libertad, sustituirla.

Pero el último caso para  $\forall$  y  $\exists$  requiere una explicación extra. Para eso veamos un ejemplo.

**Ejemplo:** Sea el término  $\alpha$ :

$$\forall x.(\exists y.(x + y = z))$$

Y la transformación  $\theta = \{z/(y - 1)\}$ . ¿Cómo es  $\alpha\theta$ ?

**Respuesta:** No importa, podemos hacer la sustitución ya que  $z$  no es ligada ¿no? Pues no, porque en efecto  $z$  no es ligada, pero será sustituida por una función de una variable ligada, de  $y$ . En este caso debemos hacer un cambio de variable para el término original, llamemos  $\tau = \{y/w\}$  **que se aplicara sobre el cuantificador y el predicado (es un caso único en el que se puede hacer esto, por eso lo ponemos entre paréntesis)**, junto con  $\theta = \{z/(y - 1)\}$ .

$$\begin{aligned} \alpha(\tau)\theta &= (\forall x.(\exists y.(x + y = z))_{[y/w]})_{[z/(y-1)]} \\ &= (\forall x.(\exists y.(x + y = z))_{[y/w]})_{[z/(y-1)]} \\ &= (\forall x.(\exists w.(x + w = z)))_{[z/(y-1)]} \\ &= (\forall x.(\exists w.(x + w = y - 1))) \end{aligned}$$

## Diferencia entre relaciones y funciones

Imaginemos que tenemos un dominio  $U$  al que pertenecen todas las variables y constantes de nuestro universo a trabajar. Las *funciones* tendrán dominio en  $U^n$  y codominio en  $U$ . Y una relación igualmente se define en  $U^n$  para cada símbolo de relación n-aria.

Pongamos un ejemplo:

- $U = \{1, 2, 3, 4\}$
- $R = \{\langle 1, 1 \rangle, \langle 1, 2 \rangle, \langle 1, 3 \rangle, \langle 1, 4 \rangle, \langle 2, 3 \rangle\}$
- $f = \{\langle 1, 2 \rangle, \langle 2, 3 \rangle, \langle 3, 4 \rangle, \langle 4, 1 \rangle\}$
- $a = 1, b = 2$

Y tendremos las siguientes fórmulas en el cálculo de predicados:

1.  $\forall x.R(x, f(x))$  falso
2.  $\exists x.R(x, f(x))$  Cierto
3.  $\forall x.R(a, x)$  Cierto
4.  $\exists x.R(b, x)$  Cierto
5.  $\forall x.[\exists y.R(x, y) \rightarrow R(x, 3)]$  Cierto (ver tabla de verdad de  $\rightarrow$ )

## 1. Satisfacción: verdad y validez

Vamos a tener una lista de sabores de helado y si están sabrosos, entonces tendremos  $\text{forall } x.S(x)$ . Tendremos un universo de interpretaciones  $U$  y las tres funciones:

$$\begin{aligned}\Psi &: \text{Var}_U C \rightarrow U \\ \Phi &: \{f_k^n\} \rightarrow \{\phi : U^n \rightarrow U\} \\ \Pi &: \{P_k^n\} \rightarrow \{R \subseteq U^n\}\end{aligned}$$

Las funciones pueden combinar más de un sabor, o incluso ponerlo en otra presentación, las relaciones se deben satisfacer.

Pero imaginemos que ahora tenemos una interpretación de cócteles de mariscos, en esa interpretación algunos valores darán verdadero en el predicado  $S(x)$ , pero no lo harán así en la interpretación de sabores de helado.

¿Y las tortillas de harina? Para que no suene tan raro nos quedamos con crepas.

- Dada una fórmula  $\sigma$  y una asignación  $s$ , la interpretación satisface a  $\sigma$  con  $s$  si

$$\sigma_U(s) = V \tag{1}$$

- Una fórmula es satisficible en el sistema de interpretación  $U$  si existe algún valor de asignación para el cual

$$\sigma_U(s) = V \tag{2}$$

- Una fórmula es válida (o verdadera) en  $U$  si

$$\sigma_U(s) = V \tag{3}$$

para toda asignación  $s$ .

**Ejemplo:** Para dar un ejemplo primero construiremos una estructura de primer orden. Definimos el lenguaje de la aritmética con  $\mathbf{CS} = \{0\}$  y  $\mathbf{CF} = \{S, +, *\}$ . Si nos plantamos en los naturales (modelo), las constantes y símbolos funcionales se interpretan de la siguiente manera:

- 0 es el 0 en los naturales,
- $S$  es la función sucesor, tal que  $S(x) = x + 1$ ,
- $+$  es la suma sobre los naturales,
- $*$  es la multiplicación sobre los naturales.

Noten que si cambiamos de modelo, si en lugar de estar en los naturales nos vamos a los racionales y a los reales, la estructura se sigue manteniendo.

Ahora si ya estamos en esa estructura y ese modelo ¿qué podemos decir de las siguientes afirmaciones?

$$\begin{aligned} & \forall x.(\neg(S(x) \doteq 0)) \\ & \forall x.(\forall y.((S(x) \doteq S(y)) \rightarrow (x \doteq y))) \\ & \quad \forall x.(x + 0 \doteq x) \\ & \forall x.(\forall y.(x + S(y) \doteq S(x + y))) \\ & \quad \forall x.(x * 0 \doteq 0) \\ & \forall x.(\forall y.(x * S(y) \doteq x * y + x)) \end{aligned}$$

Para toda fórmula  $A$  con una variable libre  $x$

$$(A(0) \wedge \forall x.(A(x) \rightarrow A(S(x)))) \rightarrow \forall y.A(y)$$

Venamos un poco las definiciones para ver si queda más claro el tema.

**Definición 10** Dado un lenguaje de primer orden  $\mathbf{L}$ , una  $\mathbf{L}$ -estructura (o simplemente una estructura)  $\mathbf{M}$  es una pareja  $\mathbf{M} = \{M, I\}$  donde  $M$  es un conjunto no vacío llamado el dominio de la estructura e  $I$  es una función llamada función de interpretación que asigna funciones y predicados en  $M$  a los símbolos  $\mathbf{L}$  de acuerdo a:

- Para todo símbolo función  $f$  de rango  $n > 0$ ,  $I(f) : M^n \rightarrow M$ , o  $f_M$ , es una función  $n$ -aria.
- Para toda constante  $c$ ,  $I(c)$ , o  $c_M$ , es un elemento de  $M$
- Para todo símbolo predicado  $P$  de rango  $n \geq 0$ ,  $I(P) : M^n \rightarrow \text{BOOL}$ , o  $P_M$ , es un predicado  $n$ -ario. En particular, los símbolos predicados de rango 0 son interpretados como valores de verdad.

Esto es un tanto abstracto, pero lo que sucede es que podemos definir un lenguaje de primer orden, pero también debemos darle sentido a lo que hacemos, para eso necesitamos de una estructura en la que podamos evaluar los términos. Tal como los ejemplos de las combinaciones de sabores de helado, podemos tener el lenguaje para armarlos, pero sólo bajo la estructura de helado (que por lo regular es frío y de sabor dulce) es que tienen sentido sólo ciertas combinaciones de sabores. Ahora veamos como es que evaluamos los términos de acuerdo a este formalismo.

**Definición 11** Dado un lenguaje  $\mathbf{L}$  de primer orden y una  $\mathbf{L}$ -estructura  $\mathbf{M}$ , una asignación es una función  $s : \mathbf{V} \rightarrow M$  del conjunto de variables  $\mathbf{V}$  al dominio  $M$ . El conjunto de todas esas funciones lo escribimos como  $[\mathbf{V} \rightarrow M]$

Asignamos un valor para eventualmente tener una respuesta de verdadero o falso, el significado de una fórmula es una función de esta asignación a los booleanos, entonces el conjunto de todas las funciones se convierte en  $[[\mathbf{V} \rightarrow M] \rightarrow \text{BOOL}]$ .

Para tratar de que quede más claro veamos un ejemplo.

**Ejemplo:** Sea el lenguaje de primer orden  $\mathbf{L} = \{=, \leq, +, *, 0, 1\}$ , una interpretación de este lenguaje es:

$$M = \{\mathbb{N}, =, \leq, +, *, \dots, 0, 1\},$$

donde  $\mathbb{N}$  es el conjunto de los números naturales, “=” es la relación de igualdad en sobre el mismo conjunto,  $\leq$  la relación “menor o igual” sobre los naturales, y “+” y “\*” son la suma y multiplicación en  $\mathbb{N}$  respectivamente.

**Teorema 1** Si  $\phi$  y  $\sigma$  son fórmulas en lógica de predicados. las siguientes fórmulas son derivadas en lógica de predicados:

- $\forall x.(\phi \rightarrow \sigma) \rightarrow \forall x.\phi \rightarrow \forall x.\sigma$
- $(\forall x.\phi \rightarrow \forall x.\sigma) \rightarrow \exists x.(\phi \rightarrow \sigma)$
- $(\exists x.\phi \rightarrow \exists x.\sigma) \rightarrow \exists x.(\phi \rightarrow \sigma)$
- $\exists x.(\phi \iff \sigma) \rightarrow \forall x.\phi \rightarrow \exists x.\sigma$
- $x.\phi \vee \forall x.\sigma \rightarrow \forall x.(\phi \vee \sigma)$
- $\forall x.(\phi \vee \sigma) \rightarrow \exists x.\phi \vee \forall x.\sigma$
- $\exists x.(\phi \vee \sigma) \iff \exists x.\phi \vee \exists x.\sigma$
- $\exists x.(\phi \wedge \sigma) \rightarrow \exists x.\phi \wedge \exists x.\sigma$
- $\forall x.(\phi \wedge \sigma) \rightarrow \forall x.\sigma \wedge \exists x.\sigma$
- $\forall x.(\phi \wedge \sigma) \iff \forall x.\phi \wedge \forall x.\sigma$
- $\exists y.(\forall x.\phi) \rightarrow \forall x.(\exists y.\phi)$
- $\forall x.(\forall y.\phi) \iff \forall y.(\forall x.\phi)$
- $\exists x.(\exists y.\phi) \iff \exists y.(\exists x.\phi)$
- $\forall x.\phi \iff \phi$  si no hay ninguna ocurrencia libre de  $x$  en  $\phi$
- $\exists x.\phi \iff \phi$  si no hay ninguna ocurrencia libre de  $x$  en  $\phi$

## Indecibilidad

### Deducción natural

La deducción natural para la lógica de predicados es una ampliación de la de la lógica de proposiciones. Si recuerdan algo de lo más relevante que se agregó fueron los cuantificadores ( $\forall, \exists$ ), ¿cómo cambia la deducción natural con estos cuantificadores? El cambio es pequeño, y aunque parezca un tanto automático, tiene mucho que ver, de nuevo, con esa palabra algo prohibida aquí: la intuición.



Partimos de un ejemplo. **Ejemplo:** Demuestra por deducción natural la consecuencia lógica de

$$\forall x.\phi(x) \rightarrow \exists x.\phi(x)$$

La intuición nos dice que si algo se cumple para toda  $x$  es obvio que existe alguna  $x$  para la que se cumple. Pero no debemos guiarnos sólo por la intuición (aunque es algo que nos puede ayudar y en ocasiones jugarnos en contra).

Hacemos la deducción natural como acostumbramos con los cuadros anidados:

|   |
|---|
| $\forall x.\phi(x)$ (Hipótesis)                               |
| $\phi(x)$ Eliminamos $\forall$ de la línea anterior           |
| $\exists x.\phi(x)$ Insertamos $\exists$ en la línea anterior |

$$\forall x.\phi(x) \rightarrow \exists x.\phi(x)$$

Como pueden ver hay unas reglas más o menos directas para introducir o eliminar los cuantificadores, para verlas en detalle podemos escribirlas como:

|   |   |
|---|---|
| $\frac{A(x)}{\forall x.A(x)} \text{ I } \forall.$ | $\frac{\forall x.A(x)}{A(x)} \text{ E } \forall$  |
| $\frac{A(x)}{\exists x.A(x)} \text{ I } \exists.$ | $\frac{\exists x. \left[ \begin{array}{c} A(x) \\ \vdots \\ B \end{array} \right]}{B} \text{ E } \exists$ |

Tiene sentido poder introducir gratuitamente el  $\forall$ , si  $A(x)$  es cierto es claro que lo estamos diciendo que es cierto para toda  $x$ . De igual manera si decimos que es cierto  $A(x)$  podemos dejar un poco la generalidad y asegurar completamente que al menos existe una  $x$  para a cual se cumple, es decir  $\exists x.A(x)$ .

Veamos algunos ejemplos más.

**Ejemplo:** Demuestra por deducción natural la consecuencia lógica

$$\forall x.A(x), \forall x.B(x) \rightarrow \forall x.(A(x) \wedge B(x))$$

**Deducción natural:**

|   |                         |                       |
|---|-------------------------|-----------------------|
| $\forall x.A(x), \forall x.B(x)$ (Hip.)   |                         |                       |
| <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;"> <math>\forall x.A(x)</math> (Hip.)           </td> </tr> <tr> <td style="padding: 5px;"> <math>A(x)</math> (E <math>\forall</math>)           </td> </tr> </table> | $\forall x.A(x)$ (Hip.) | $A(x)$ (E $\forall$ ) |
| $\forall x.A(x)$ (Hip.)   |                         |                       |
| $A(x)$ (E $\forall$ )   |                         |                       |
| $A(x)$  |                         |                       |
| <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 5px;"> <math>\forall x.B(x)</math> (Hip.)           </td> </tr> <tr> <td style="padding: 5px;"> <math>B(x)</math> (E <math>\forall</math>)           </td> </tr> </table> | $\forall x.B(x)$ (Hip.) | $B(x)$ (E $\forall$ ) |
| $\forall x.B(x)$ (Hip.)   |                         |                       |
| $B(x)$ (E $\forall$ )   |                         |                       |
| $B(x)$  |                         |                       |
| $A(x) \wedge B(x)$ (I $\wedge$ )  |                         |                       |
| $\forall x.(A(x) \wedge B(x))$ (I $\forall$ )   |                         |                       |
| $\forall x.A(x), \forall x.B(x) \rightarrow \forall x.(A(x) \wedge B(x))$ (I $\rightarrow$ )  |                         |                       |

**Ejemplo:** Demuestra por deducción natural la consecuencia lógica de:

$$\exists x.(A(x) \wedge B(x)) \rightarrow \exists x.A(x)$$

**Reaspueta:**

|  |
|--|
| $\exists x.(A(x) \wedge B(x))$ (Hip.)<br>$A(x) \wedge B(x)$ (E $\exists$ )                       |
| $A(x)$ (E $\wedge$ )<br>$\exists x.A(x)$   |
| $\exists x.A(x)$<br>$\exists x.(A(x) \wedge B(x)) \rightarrow \exists x.A(x)$ (I $\rightarrow$ ) |

**Ejemplo:** Demuestra por deducción natural la consecuencia lógica de:

$$\forall x.(A(x) \rightarrow \neg B(x)) \rightarrow \neg \exists x.(A(x) \wedge B(x))$$

**Respuesta:**

|   |
|---|
| $\forall x.(A(x) \rightarrow \neg B(x))$ (Hip.)<br>$A(x) \rightarrow \neg B(x)$ (E $\forall$ )  |
| $\exists x.(A(x) \wedge B(x))$ (Hip.)<br>$A(x) \wedge B(x)$ (E $\exists$ )<br>$A(x)$ E $\wedge$<br>$\neg B(x)$ (Hip.)<br>$\perp$ (Reducción al absurdo) |
| $\neg \exists x.(A(x) \wedge B(x))$<br>$\forall x.(A(x) \rightarrow \neg B(x)) \rightarrow \neg \exists x.(A(x) \wedge B(x))$ (I $\rightarrow$ )        |

## Referencias

- [1] Metakides, G. y Nerode, A. "Principles of Logic and Logic Programming". Elsevier, Amsterdam (1996).
- [2] Reeves, S. y Clarke, M. "Logic for computer science" Addison-Wesley Publishers Ltd. (1990)