

Lógica informática (2005–06)

Tema 6: Sintaxis y semántica de la lógica de primer orden

José A. Alonso Jiménez
 Grupo de Lógica Computacional
 Dpto. Ciencias de la Computación e Inteligencia Artificial
 Universidad de Sevilla

1

Limitación expresiva de la lógica proposicional

- Ejemplo 1: Si Sevilla es vecina de Cádiz, entonces Cádiz es vecina de Sevilla. Sevilla es vecina de Cádiz. Por tanto, Cádiz es vecina de Sevilla
 - Representación en lógica proposicional:

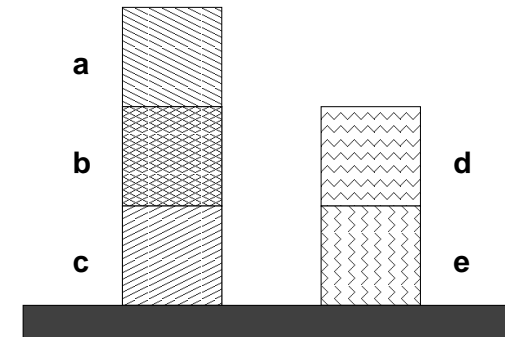
$$\{SvC \rightarrow CvS, SvC\} \models CvS$$
- Ejemplo 2: Si una ciudad es vecina de otra, entonces la segunda es vecina de la primera. Sevilla es vecina de Cádiz. Por tanto, Cádiz es vecina de Sevilla
 - Representación en lógica proposicional: Imposible
 - Representación en lógica de primer orden:

$$\{(\forall x)(\forall y)[vecina(x,y) \rightarrow vecina(y,x)], vecina(Sevilla,Cadiz)\} \\ \models vecina(Cadiz,Sevilla)$$

2

Potencia expresiva de la lógica de primer orden

Mundo de los bloques



- Simbolización:
 - sobre(x,y) se verifica si el bloque x está colocado sobre el bloque y
 - sobre_mesa(x) se verifica si el bloque x está sobre la mesa
- Situación del ejemplo:

$$\text{sobre}(a,b), \text{sobre}(b,c), \text{sobre_mesa}(c), \text{sobre}(d,e), \text{sobre_mesa}(e)$$

3

Potencia expresiva de la lógica de primer orden

- Definiciones:
 - bajo(x,y) se verifica si el bloque x está debajo del bloque y

$$(\forall x)(\forall y)[\text{bajo}(x,y) \leftrightarrow \text{sobre}(y,x)]$$
 - encima(x,y) se verifica si el bloque x está encima del bloque y pudiendo haber otros bloques entre ellos

$$(\forall x)(\forall y)[\text{encima}(x,y) \leftrightarrow \text{sobre}(x,y) \vee (\exists z)[\text{sobre}(x,z) \wedge \text{encima}(z,y)]]$$
 - libre(x) se verifica si el bloque x no tiene bloques encima

$$(\forall x)[\text{libre}(x) \leftrightarrow \neg(\exists y) \text{sobre}(y,x)]$$
 - pila(x,y,z) se verifica si el bloque x está sobre el y , el y sobre el z y el z sobre la mesa

$$(\forall x)(\forall y)(\forall z)[\text{pila}(x,y,z) \leftrightarrow \text{sobre}(x,y) \wedge \text{sobre}(y,z) \wedge \text{sobre_mesa}(z)]$$
- Propiedades:
 - Si z,y,z es una pila entonces y no está libre

$$(\forall x)(\forall y)(\forall z)[\text{pila}(x,y,z) \rightarrow \neg \text{libre}(y)]$$

4

Potencia expresiva de la lógica de primer orden

Representación del mundo de los bloques con funciones e igualdad:

- Simbolización:
 - ▶ $\text{es_bloque}(x)$ se verifica si x es un bloque.
 - ▶ $\text{superior}(x)$ es el bloque que está sobre el bloque x .
- Situación del ejemplo:
 $\text{es_bloque}(a), \text{es_bloque}(b), \text{es_bloque}(c), \text{es_bloque}(d), \text{es_bloque}(e)$
 $\text{superior}(b) = a, \text{superior}(c) = b, \text{superior}(e) = d$
- Definiciones:
 - ▶ $\text{sobre_mesa}(x)$ se verifica si el bloque x está sobre la mesa
 $(\forall x)[\text{sobre_mesa}(x) \leftrightarrow \text{es_bloque}(x) \wedge \neg(\exists y) \text{superior}(y) = x]$
 - ▶ $\text{libre}(x)$ se verifica si el bloque x no tiene bloques encima
 $(\forall x)[\text{libre}(x) \leftrightarrow \neg(\exists y) \text{superior}(x) = y]$
 - ▶ $\text{tope}(x)$ es el bloque libre que está encima de x
 $(\forall x)[(\text{libre}(x) \rightarrow \text{tope}(x) = x) \wedge$
 $(\neg \text{libre}(x) \rightarrow \text{tope}(x) = \text{tope}(\text{superior}(x)))]$

5

Lenguaje de primer orden

- Símbolos lógicos:
 - ▶ Variables: $x, y, z, \dots, x_1, x_2, \dots$
 - ▶ Conectivas: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$.
 - ▶ Cuantificadores: \forall, \exists .
 - ▶ Símbolo de igualdad: $=$.
- Símbolos propios:
 - ▶ Símbolos de constantes: $a, b, c, \dots, a_1, a_2, \dots$
 - ▶ Símbolos de predicado (con aridad): $P, Q, R, \dots, P_1, P_2, \dots$
 - ▶ Símbolos de función (con aridad): $f, g, h, \dots, f_1, f_2, \dots$
- Símbolos auxiliares: “(”, “)”, “,”, “.”.
- Notación:
 - ▶ L, L_1, L_2, \dots representan lenguajes de primer orden.
 - ▶ Var representa el conjunto de las variables.
- Los símbolos de predicados de aridad mayor que 1 se llaman de relaciones.

6

Ejemplos de lenguajes de primer orden

- Lenguaje del mundo de los bloques:
 - ▶ Símbolos de constantes: a, b, c, d, e
 - ▶ Símbolos de predicado (y de relación):
 - de aridad 1: $\text{sobre_mesa}, \text{libre}, \text{es_bloque}$
 - de aridad 2: $\text{sobre}, \text{bajo}, \text{encima}$
 - de aridad 3: pila
 - ▶ Símbolos de función (de aridad 1): $\text{superior}, \text{tope}$
- Lenguaje de la aritmética:
 - ▶ Símbolos de constantes: $0, 1$
 - ▶ Símbolos de función:
 - monaria: s (siguiente)
 - binarias: $+, \cdot$
 - ▶ Símbolo de predicado binario: $<$

7

Términos

- Def. de **término** de un lenguaje de primer orden L :
 - ▶ Las variables son términos de L .
 - ▶ Las constantes de L son términos de L .
 - ▶ Si f es un símbolo de función n -aria de L y t_1, \dots, t_n son términos de L , entonces $f(t_1, \dots, t_n)$ es un término de L .
- Ejemplos:
 - ▶ En el lenguaje de la aritmética,
 - $+(\cdot(x, 1), s(y))$ es un término, que se suele escribir como $(x \cdot 1) + s(y)$
 - $+(\cdot(x, <), s(y))$ no es un término
 - ▶ En el lenguaje del mundo de los bloques,
 - $\text{superior}(\text{superior}(c))$ es un término.
 - $\text{libre}(\text{superior}(c))$ no es un término.
- Notación:
 - ▶ s, t, t_1, t_2, \dots representan términos.
 - ▶ $\text{Térm}(L)$ representa el conjunto de los términos de L

8

Fórmulas atómicas

- Def. de **fórmula atómica** de un lenguaje de primer orden L :
 - ▶ Si t_1 y t_2 son términos de L , entonces $t_1 = t_2$ es una fórmula atómica de L .
 - ▶ Si P es un símbolo de relación n -aria de L y t_1, \dots, t_n son términos de L , entonces $P(t_1, \dots, t_n)$ es una fórmula atómica de L .
- Ejemplos:
 - ▶ En el lenguaje de la aritmética,
 - $<(\cdot(x, 1), s(y))$ es una fórmula atómica que se suele escribir como $x \cdot 1 < s(y)$
 - $+(x, y) = \cdot(x, y)$ es una fórmula atómica que se suele escribir como $x + y = x \cdot y$
 - ▶ En el lenguaje del mundo de los bloques,
 - $\text{libre}(\text{superior}(c))$ es una fórmula atómica.
 - $\text{tope}(c) = \text{superior}(b)$ es una fórmula atómica.
- Notación:
 - ▶ A, B, A_1, A_2, \dots representan fórmulas atómicas.
 - ▶ $\text{Atom}(L)$ representa el conjunto de las fórmulas atómicas de L .

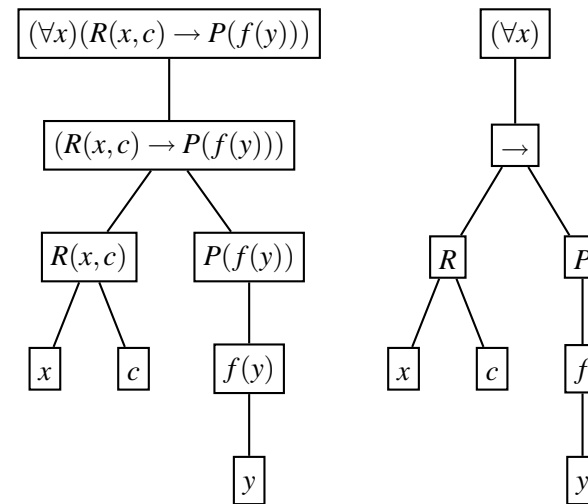
9

Fórmulas

- Definición de las **fórmulas** de L :
 - ▶ Las fórmulas atómicas de L son fórmulas de L .
 - ▶ Si F y G son fórmulas de L , entonces $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \rightarrow G)$ y $(F \leftrightarrow G)$ son fórmulas de L .
 - ▶ Si F es una fórmula de L , entonces $(\forall x)F$ y $(\exists x)F$ son fórmulas de L .
- Ejemplos:
 - ▶ En el lenguaje de la aritmética,
 - $(\forall x)(\exists y) <(x, y)$ es una fórmula que se escribe como $(\forall x)(\exists y)x < y$
 - $(\forall x)(\exists y) + (x, y)$ no es una fórmula.
 - ▶ En el lenguaje del mundo de los bloques,
 - $(\forall x)(\text{tope}(x) = x \leftrightarrow \text{libre}(x))$ es una fórmula.
- Notación:
 - ▶ F, G, H, F_1, F_2, \dots representan fórmulas.
 - ▶ $\text{Fórm}(L)$ representa el conjunto de las fórmulas de L .

10

Árboles de análisis (o de formación)



11

Subfórmulas

- Def: El conjunto $\text{Subf}(F)$ de las **subfórmulas** de una fórmula F se define recursivamente por:

$$\text{Subf}(F) = \begin{cases} \{F\}, & \text{si } F \text{ es una fórmula atómica;} \\ \{F\} \cup \text{Subf}(G), & \text{si } F = \neg G; \\ \{F\} \cup \text{Subf}(G) \cup \text{Subf}(H), & \text{si } F = G * H; \\ \{F\} \cup \text{Subf}(G), & \text{si } F = (\forall x)G; \\ \{F\} \cup \text{Subf}(G), & \text{si } F = (\exists x)G \end{cases}$$
- Ejemplo:

$$\text{Subf}((\forall x)(R(x, c) \rightarrow P(f(y)))) = \{ (\forall x)(R(x, c) \rightarrow P(f(y))), (R(x, c) \rightarrow P(f(y))), R(x, c), P(f(y)) \}$$

12

Criterios de reducción de paréntesis

- Pueden eliminarse los paréntesis externos.
 $F \wedge G$ es una abreviatura de $(F \wedge G)$
- Precedencia de asociación de conectivas y cuantificadores: $\forall, \exists, \neg, \wedge, \vee, \rightarrow, \leftrightarrow$.
 $\forall x P(x) \rightarrow Q(x)$ es una abreviatura de $((\forall x)P(x)) \rightarrow Q(x)$
- Cuando una conectiva se usa repetidamente, se asocia por la derecha.
 $F \vee G \vee H$ es una abreviatura de $(F \vee (G \vee H))$
 $F \wedge G \wedge H \rightarrow \neg F \vee G$ es una abreviatura de $((F \wedge (G \wedge H)) \rightarrow (\neg F \vee G))$
- Los símbolos binarios pueden escribirse en notación infija.
 $x + y$ es una abreviatura de $+(x, y)$
 $x < y$ es una abreviatura de $<(x, y)$

13

Conjuntos de variables

- Def.: El **conjunto de las variables** del término t se define recursivamente por:

$$V(t) = \begin{cases} \emptyset, & \text{si } t \text{ es una constante;} \\ \{x\}, & \text{si } t \text{ es una variable } x; \\ V(t_1) \cup \dots \cup V(t_n), & \text{si } t \text{ es } f(t_1, \dots, t_n) \end{cases}$$
- Def.: El **conjunto de las variables** de la fórmula F se define recursivamente por:

$$V(F) = \begin{cases} V(t_1) \cup V(t_2), & \text{si } F \text{ es } t_1 = t_2; \\ V(t_1) \cup \dots \cup V(t_n), & \text{si } F \text{ es } P(t_1, \dots, t_n); \\ V(G), & \text{si } F \text{ es } \neg G; \\ V(G) \cup V(H), & \text{si } F \text{ es } G * H; \\ V(G), & \text{si } F \text{ es } (\forall x)G; \\ V(G), & \text{si } F \text{ es } (\exists x)G \end{cases}$$
- Ejemplos:
 - El conjunto de las variables de $(\forall x)(R(x, c) \rightarrow P(f(y)))$ es $\{x, y\}$.
 - El conjunto de las variables de $(\forall x)(R(a, c) \rightarrow P(f(y)))$ es $\{y\}$.

14

Apariciones libres y ligadas

- Def.: Una aparición (u ocurrencia) de la variable x en la fórmula F es **ligada** si es en una subfórmula de F de la forma $(\forall x)G$ ó $(\exists x)G$.
- Def.: Una aparición (u ocurrencia) de la variable x en la fórmula F es **libre** si no es ligada
- Ejemplo: Las apariciones ligadas son las subrayadas:

$$(\forall x)(P(\underline{x}) \rightarrow R(\underline{x}, y)) \rightarrow ((\exists y)P(\underline{y}) \rightarrow R(z, x))$$

$$(\exists x)R(x, y) \vee (\forall y)P(y)$$

$$(\forall x)(P(\underline{x}) \rightarrow (\exists y)R(\underline{x}, \underline{y}))$$

$$P(x) \rightarrow R(x, y)$$

15

Variables libres y ligadas

- Def.: La variable x es **libre** en F si tiene una aparición libre en F .
- Def.: La variable x es **ligada** en F si tiene una aparición ligada en F .
- Prop.: El **conjunto de las variables libres** de una fórmula F es:

$$VL(F) = \begin{cases} V(t_1) \cup V(t_2), & \text{si } F \text{ es } t_1 = t_2; \\ V(t_1) \cup \dots \cup V(t_n), & \text{si } F \text{ es } P(t_1, \dots, t_n); \\ VL(G), & \text{si } F \text{ es } \neg G; \\ VL(G) \cup VL(H), & \text{si } F \text{ es } G * H; \\ VL(G) \setminus \{x\}, & \text{si } F \text{ es } (\forall x)G; \\ VL(G) \setminus \{x\}, & \text{si } F \text{ es } (\exists x)G \end{cases}$$
- Ejemplo:

Fórmula	Ligadas	Libres
$(\forall x)(P(x) \rightarrow R(x, y)) \rightarrow ((\exists y)P(y) \rightarrow R(x, z))$	x, y	x, y, z
$(\forall x)(P(x) \rightarrow (\exists y)R(x, y))$	x, y	
$(\forall z)(P(x) \rightarrow R(x, y))$		x, y

16

Fórmulas cerradas y abiertas

- Fórmula cerradas:
 - ▶ Def.: Una **fórmula cerrada** (o **sentencia**) es una fórmula sin variables libres.
 - ▶ Ejemplos: $(\forall x)(P(x) \rightarrow (\exists y)R(x,y))$ es cerrada.
 $(\exists x)R(x,y) \vee (\forall y)P(y)$ no es cerrada.
- Fórmulas abiertas:
 - ▶ Def.: Una **fórmula abierta** es una fórmula con variables libres.
 - ▶ Ejemplos: $(\forall x)(P(x) \rightarrow (\exists y)R(x,y))$ no es abierta.
 $(\exists x)R(x,y) \vee (\forall y)P(y)$ es abierta.

17

Semántica: Estructuras, asignaciones e interpretaciones

- Una **estructura del lenguaje L** es un par $\mathcal{I} = (U, I)$ tal que:
 - ▶ U es un conjunto no vacío, denominado **universo** de la estructura;
 - ▶ I es una función con dominio el conjunto de símbolos propios de L tal que
 - si c es una constante de L , entonces $I(c) \in U$;
 - si f es un símbolo de función n -aria de L , entonces $I(f) : U^n \rightarrow U$;
 - si P es un símbolo de relación 0-aria de L , entonces $I(P) \in \{1, 0\}$;
 - si R es un símbolo de relación n -aria ($n > 0$) de L , entonces $I(R) \subseteq U^n$;
- Una **asignación A en una estructura (U, I)** es una función $A : \text{Var} \rightarrow U$ que hace corresponder a cada variable del alfabeto un elemento del universo de la estructura.
- Una **interpretación de L** es un par (\mathcal{I}, A) formado por una estructura \mathcal{I} de L y una asignación A en \mathcal{I} .
- Notación: A veces se usa para los valores de verdad **V** y **F** en lugar de 1 y 0.

18

Ejemplos de estructuras

Sea L el lenguaje de la aritmética cuyos símbolos propios son:

constante: 0;
 símbolo de función monaria: s ;
 símbolo de función binaria: $+$ y
 símbolo de relación binaria: \leq

- Primera estructura de L :

$U_1 = \mathbb{N}$
 $I_1(0) = 0$
 $I_1(s) = \{(n, n+1) : n \in \mathbb{N}\}$ (sucesor)
 $I_1(+)$ = $\{(a, b, a+b) : a, b \in \mathbb{N}\}$ (suma)
 $I_1(\leq) = \{(n, m) : n, m \in \mathbb{N}, n \leq m\}$ (menor o igual)

- Segunda estructura de L :

$U_2 = \{0, 1\}^*$ (cadenas de 0 y 1)
 $I_2(0) = \varepsilon$ (cadena vacía)
 $I_2(s) = \{(w, w1) : w \in \{0, 1\}^*\}$ (siguiente)
 $I_2(+)$ = $\{(w_1, w_2, w_1w_2) : w_1, w_2 \in \{0, 1\}^*\}$ (concatenación)
 $I_2(\leq) = \{(w_1, w_2) : w_1, w_2 \in \{0, 1\}^*, w_1 \text{ es prefijo de } w_2\}$ (prefijo)

19

Ejemplos de estructuras

- Tercera estructura de L :

$U_3 = \{\text{abierto}, \text{cerrado}\}$
 $I_3(0) = \text{cerrado}$
 $I_3(s) = \{(\text{abierto}, \text{cerrado}), (\text{cerrado}, \text{abierto})\}$
 $I_3(+)$ = $\{(\text{abierto}, \text{abierto}, \text{abierto}), (\text{abierto}, \text{cerrado}, \text{abierto}),$
 $(\text{cerrado}, \text{abierto}, \text{abierto}), (\text{cerrado}, \text{cerrado}, \text{cerrado})\}$
 $I_3(\leq) = \{(\text{abierto}, \text{abierto}), (\text{cerrado}, \text{abierto}), (\text{cerrado}, \text{cerrado})\}$

e	$I_3(s)(e)$		$I_3(\leq)$	abierto	cerrado
abierto	cerrado		abierto	1	0
cerrado	abierto		cerrado	1	1
$I_3(+)$	abierto	cerrado	abierto	1	0
	abierto	abierto	abierto	1	0
	abierto	cerrado	cerrado	1	1
	cerrado	abierto	abierto	1	0
	cerrado	cerrado	cerrado	1	1

20

Ejemplo de evaluación de términos

- Sean L el lenguaje de la página 19 y t el término $s(x+s(0))$.
 - Si \mathcal{I} es la primera estructura y $A(x) = 3$, entonces

$$\begin{aligned}\mathcal{I}_A(t) &= \mathcal{I}_A(s(x+s(0))) = s^I(3 +^I s^I(0^I)) = \\ &= s^I(3 +^I s^I(0)) = s^I(3 +^I 1) = \\ &= s^I(4) = 5\end{aligned}$$
 - Si \mathcal{I} es la segunda estructura y $A(x) = 10$, entonces

$$\begin{aligned}\mathcal{I}_A(t) &= \mathcal{I}_A(s(x+s(0))) = s^I(10 +^I s^I(0^I)) = \\ &= s^I(10 +^I s^I(\varepsilon)) = s^I(10 +^I 1) = \\ &= s^I(101) = 1011\end{aligned}$$
 - Si \mathcal{I} es la tercera estructura y $A(x) = \text{abierto}$, entonces

$$\begin{aligned}\mathcal{I}_A(t) &= \mathcal{I}_A(s(x+s(0))) = s^I(\text{abierto} +^I s^I(0^I)) = \\ &= s^I(\text{abierto} +^I s^I(\text{cerrado})) = s^I(\text{abierto} +^I \text{abierto}) = \\ &= s^I(\text{abierto}) = \text{cerrado}\end{aligned}$$

21

Evaluación de términos

- Def.: Dada una estructura $\mathcal{I} = (U, I)$ de L y una asignación A en \mathcal{I} , se define la **función de evaluación de términos** $\mathcal{I}_A : \text{Térm}(L) \rightarrow U$ por

$$\mathcal{I}_A(t) = \begin{cases} I(c), & \text{si } t \text{ es una constante } c; \\ A(x), & \text{si } t \text{ es una variable } x; \\ I(f)(\mathcal{I}_A(t_1), \dots, \mathcal{I}_A(t_n)), & \text{si } t \text{ es } f(t_1, \dots, t_n) \end{cases}$$

- $\mathcal{I}_A(t)$ se lee “el valor de t en \mathcal{I} respecto de A ”.
- Ejemplo: Sean L el lenguaje de la página 19, t el término $s(+ (x, s(0)))$, \mathcal{I} la primera estructura y $A(x) = 3$.

$$\begin{aligned}\mathcal{I}_A(t) &= \mathcal{I}_A(s(+ (x, s(0)))) &&= I(s)(\mathcal{I}_A(+ (x, s(0)))) = \\ &= I(s)(I(+)(\mathcal{I}_A(x), \mathcal{I}_A(s(0)))) &&= I(s)(I(+)(A(x), \mathcal{I}_A(s(0)))) = \\ &= I(s)(I(+)(3, I(s)(\mathcal{I}_A(0)))) &&= I(s)(I(+)(3, I(s)(I(0)))) = \\ &= I(s)(I(+)(3, I(s)(0))) &&= I(s)(I(+)(3, 1)) = \\ &= I(s)(4) &&= 5\end{aligned}$$

22

Evaluación de fórmulas

- Def.: Dada una estructura $\mathcal{I} = (U, I)$ de L y una asignación A sobre \mathcal{I} , se define la **función de evaluación de fórmulas** $\mathcal{I}_A : \text{Fórm}(L) \rightarrow \mathbb{B}$ por
 - Si F es $t_1 = t_2$, $\mathcal{I}_A(F) = H_=(\mathcal{I}_A(t_1), I_A(t_2))$
 - Si F es $P(t_1, \dots, t_n)$, $\mathcal{I}_A(F) = H_{I(P)}(\mathcal{I}_A(t_1), \dots, \mathcal{I}_A(t_n))$
 - Si F es $\neg G$, $\mathcal{I}_A(F) = H_{\neg}(\mathcal{I}_A(G))$
 - Si F es $G * H$, $\mathcal{I}_A(F) = H_*(\mathcal{I}_A(G), \mathcal{I}_A(H))$
 - Si F es $(\forall x)G$, $\mathcal{I}_A(F) = \begin{cases} 1, & \text{si para todo } u \in U \text{ se tiene} \\ & \mathcal{I}_{A[x/u]}(G) = 1; \\ 0, & \text{en caso contrario} \end{cases}$
 - Si F es $(\exists x)G$, $\mathcal{I}_A(F) = \begin{cases} 1, & \text{si existe algún } u \in U \text{ tal que} \\ & \mathcal{I}_{A[x/u]}(G) = 1; \\ 0, & \text{en caso contrario} \end{cases}$
- $\mathcal{I}_A(F)$ se lee “el valor de F en \mathcal{I} respecto de A ”.

23

Conceptos auxiliares para la evaluación de fórmulas

- La **función de verdad de la igualdad** en U es la función $H_= : U^2 \rightarrow \mathbb{B}$ definida por

$$H_=(u_1, u_2) = \begin{cases} 1, & \text{si } u_1 = u_2; \\ 0, & \text{en caso contrario} \end{cases}$$

- Función de verdad de una relación:** Si R es una relación n -aria en U (i.e. $R \subseteq U^n$), entonces la **función de verdad de R** es la función $H_R : U^n \rightarrow \mathbb{B}$ definida por

$$H_R(u_1, \dots, u_n) = \begin{cases} 1, & \text{si } (u_1, \dots, u_n) \in R; \\ 0, & \text{en caso contrario} \end{cases}$$

- Variante de una asignación:** Sea A una asignación en la estructura (U, I) y $u \in U$. Mediante $A[x/u]$ se representa la asignación definida por

$$A[x/u](y) = \begin{cases} u, & \text{si } y \text{ es } x; \\ A(y) & \text{si } y \text{ es distinta de } x \end{cases}$$

24

Ejemplo de evaluación de fórmula

Evaluación de $(\forall x)(\exists y)P(x,y)$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$ e $I(P) = \{(1, 1), (2, 2)\}$

$$\mathcal{I}_A((\forall x)(\exists y)P(x,y)) = V \Leftrightarrow \mathcal{I}_{A[x/1]}((\exists y)P(x,y)) = V \text{ y } \mathcal{I}_{A[x/2]}((\exists y)P(x,y)) = V$$

$$\begin{aligned} \mathcal{I}_{A[x/1]}((\exists y)P(x,y)) &= V \Leftrightarrow \mathcal{I}_{A[x/1,y/1]}P(x,y) = V \text{ ó } \mathcal{I}_{A[x/1,y/2]}P(x,y) = V \\ \mathcal{I}_{A[x/1,y/1]}P(x,y) &= P^I(1, 1) \\ &= V \end{aligned}$$

$$\text{Luego, } \mathcal{I}_{A[x/1]}((\exists y)P(x,y)) = V.$$

$$\begin{aligned} \mathcal{I}_{A[x/2]}((\exists y)P(x,y)) &= V \Leftrightarrow \mathcal{I}_{A[x/2,y/1]}P(x,y) = V \text{ ó } \mathcal{I}_{A[x/2,y/2]}P(x,y) = V \\ \mathcal{I}_{A[x/2,y/2]}P(x,y) &= P^I(2, 2) \\ &= V \end{aligned}$$

$$\text{Luego, } \mathcal{I}_{A[x/2]}((\exists y)P(x,y)) = V.$$

$$\text{Por tanto, } \mathcal{I}_A((\forall x)(\exists y)P(x,y)) = V$$

25

Ejemplo de evaluación de fórmulas

Ejemplo: Evaluación de $(\forall x)g(g(x)) = x$ en la estructura $\mathcal{I} = (U, I)$ respecto de la asignación A tales que $U = \{1, 2\}$, $I(a) = 1$, $I(g) = \{(1, 2), (2, 1)\}$, $I(P) = \{2\}$ e $I(Q) = \{(1, 2), (2, 2)\}$.

$$\mathcal{I}_A((\forall x)g(g(x)) = x) = V \Leftrightarrow \mathcal{I}_{A[x/1]}g(g(x)) = x = V \text{ y } \mathcal{I}_{A[x/2]}g(g(x)) = x = V$$

$$\begin{aligned} \mathcal{I}_{A[x/1]}(g(g(x)) = x) &= (g^I(g^I(1)) = 1) \\ &= (g^I(2) = 1) \\ &= (1 = 1) \\ &= V \end{aligned}$$

$$\begin{aligned} \mathcal{I}_{A[x/2]}(g(g(x)) = x) &= (g^I(g^I(2)) = 2) \\ &= (g^I(1) = 2) \\ &= (2 = 2) \\ &= V \end{aligned}$$

$$\text{Por tanto, } \mathcal{I}_A((\forall x)g(g(x)) = x) = V.$$

26

Dependencias en la evaluación de fórmulas

- Ejemplo de dependencia del universo: Sea G la fórmula $(\forall x)(\exists y)R(y,x)$, entonces
 - ▶ $\mathcal{I}_A(G) = V$, siendo $\mathcal{I} = (\mathbb{Z}, I), I(R) = <$ y A una asignación en \mathcal{I} .
 - ▶ $\mathcal{I}_A(G) = F$, siendo $\mathcal{I} = (\mathbb{N}, I), I(R) = <$ y A una asignación en \mathcal{I} .
- Ejemplo de dependencia de la estructura: Sea G la fórmula $(\exists x)(\forall y)R(x,y)$, entonces
 - ▶ $\mathcal{I}_A(G) = V$, siendo $\mathcal{I} = (\mathbb{N}, I), I(R) = \leq$ y A una asignación en \mathcal{I} .
 - ▶ $\mathcal{I}_A(G) = F$, siendo $\mathcal{I} = (\mathbb{N}, I), I(R) = \geq$ y A una asignación en \mathcal{I} .
- Ejemplo de dependencia de la asignación: Sea G la fórmula $(\forall y)R(x,y)$, entonces
 - ▶ $\mathcal{I}_A(G) = V$, siendo $\mathcal{I} = (\mathbb{N}, I), I(R) = \leq$ y A una asignación en \mathcal{I} tal que $A(x) = 0$.
 - ▶ $\mathcal{I}_A(G) = F$, siendo $\mathcal{I} = (\mathbb{N}, I), I(R) = \leq$ y A una asignación en \mathcal{I} tal que $A(x) = 5$.

27

Evaluación y variables libres

- Sea t un término de L e \mathcal{I} una estructura de L .
 - ▶ Si A y B son dos asignaciones en \mathcal{I} que coinciden sobre las variables de t , entonces $\mathcal{I}_A(t) = \mathcal{I}_B(t)$.
 - ▶ Si t no tiene variables, entonces $\mathcal{I}_A(t) = \mathcal{I}_B(t)$ para cualesquiera asignaciones A y B en \mathcal{I} . Se suele escribir simplemente $\mathcal{I}(t)$.
- Sea F una fórmula de L e \mathcal{I} una estructura de L .
 - ▶ Si A y B son dos asignaciones en \mathcal{I} que coinciden sobre las variables libres de F , entonces $\mathcal{I}_A(F) = \mathcal{I}_B(F)$.
 - ▶ Si F es cerrada, entonces $\mathcal{I}_A(F) = \mathcal{I}_B(F)$ para cualesquiera asignaciones A y B en \mathcal{I} . Se suele escribir simplemente $\mathcal{I}(F)$.

28

Modelo de una fórmula

- Sean F una fórmula de L e \mathcal{I} una estructura de L .
 - (\mathcal{I}, A) es una realización de F si A una asignación en \mathcal{I} tal que $\mathcal{I}_A(F) = 1$. Se representa por $\mathcal{I}_A \models F$.
 - \mathcal{I} es un modelo de F si, para toda asignación A en \mathcal{I} , $\mathcal{I}_A(F) = 1$. Se representa por $\mathcal{I} \models F$.
- Ejemplos: Sea $\mathcal{I} = (\mathbb{N}, I)$ una estructura tal que $I(f) = +$ e $I(g) = *$.
 - Si A es una asignación en \mathcal{I} tal que $A(x) = A(y) = 2$. entonces $\mathcal{I}_A \models f(x, y) = g(x, y)$,
 - Si B es una asignación en \mathcal{I} tal que $B(x) = 1, B(y) = 2$. entonces $\mathcal{I}_B \not\models f(x, y) = g(x, y)$,
 - $I \not\models f(x, y) = g(x, y)$
 - $I \models f(x, y) = f(y, x)$

29

Satisfacibilidad y validez

- Def.: Sea F una fórmula de L .
 - F es válida si toda estructura de L es modelo de F (i.e. para toda estructura \mathcal{I} de L y toda asignación A en \mathcal{I} se tiene que $\mathcal{I}_A(F) = 1$). Se representa por $\models F$.
 - F es satisfacible si tiene alguna realización (i.e. existe alguna estructura \mathcal{I} de L y alguna asignación A en I tales que $\mathcal{I}_A(F) = 1$).
 - F es insatisfacible si no tiene ninguna realización (i.e. para toda estructura \mathcal{I} de L y toda asignación A en I se tiene que $\mathcal{I}_A(F) = 0$).
- Ejemplos:
 - $(\exists x)P(x) \vee (\forall x)\neg P(x)$ es válida.
 - $(\exists x)P(x) \wedge (\exists x)\neg P(x)$ es satisfacible, pero no es válida. $\mathcal{I}_A((\exists x)P(x) \wedge (\exists x)\neg P(x)) = 1$, siendo $\mathcal{I} = (\{a, b\}, I)$, $I(P) = \{a\}$. $\mathcal{I}_A((\exists x)P(x) \wedge (\exists x)\neg P(x)) = 1$, siendo $\mathcal{I} = (\{a\}, I)$, $I(P) = \{a\}$.
 - $(\forall x)P(x) \wedge (\exists x)\neg P(x)$ es insatisfacible.

30

Satisfacibilidad y validez

- Prop.: F es válida syss $\neg F$ es insatisfacible.
 - F es válida \iff para toda estructura \mathcal{I} y toda asignación A se tiene que $\mathcal{I}_A(F) = 1$
 - \iff para toda estructura \mathcal{I} y toda asignación A se tiene que $\mathcal{I}_A(\neg F) = 0$
 - $\iff \neg F$ es insatisfacible.
- Si F es válida, entonces F es satisfacible.
 - F es válida \implies para toda estructura \mathcal{I} y toda asignación A se tiene que $\mathcal{I}_A(F) = 1$
 - \implies existe una estructura \mathcal{I} y una asignación A tales que $\mathcal{I}_A(F) = 1$
 - $\implies F$ es satisfacible.
- F es satisfacible $\not\iff \neg F$ es insatisfacible. $(\forall x)P(x)$ y $\neg(\forall x)P(x)$ son satisfacibles.
- Sea F una fórmula de L y x_1, \dots, x_n las variables libres de F .
 - F es válida syss $(\forall x_1) \dots (\forall x_n)F$ es válida. [Cierre universal].
 - F es satisfacible syss $(\exists x_1) \dots (\exists x_n)F$ es satisfacible. [Cierre existencial].

31

Modelo de un conjunto de fórmulas

- Notación: S, S_1, S_2, \dots representarán conjuntos de fórmulas.
- Def.: Sean S un conjunto de fórmulas de L e \mathcal{I} una estructura de L y A una asignación en \mathcal{I} .
 - (\mathcal{I}, A) es una realización de S si A es una asignación en \mathcal{I} tal que para toda $F \in S$ se tiene que $\mathcal{I}_A(F) = 1$. Se representa por $\mathcal{I}_A \models S$.
 - \mathcal{I} es un modelo de S si para toda $F \in S$ se tiene que $\mathcal{I} \models F$ (i.e. para toda $F \in S$ y toda asignación A en \mathcal{I} se tiene $\mathcal{I}_A(F) = 1$). Se representa por $\mathcal{I} \models S$.
- Ejemplos:
 - Sea $S = \{(\forall y)R(x, y), (\forall y)f(x, y) = y\}$.
 - (\mathcal{I}, A) con $\mathcal{I} = (\mathbb{N}, I)$, $R^I = \leq$, $f^I = +$, $A(x) = 0$ es realización de S .
 - (\mathcal{I}, A) con $\mathcal{I} = (\mathbb{N}, I)$, $R^I = <$, $f^I = +$, $A(x) = 0$ no es realización de S .
 - Sea $S = \{R(e, y), f(e, y) = y\}$.
 - $\mathcal{I} = (\mathbb{N}, I)$ con $R^I = \leq$, $f^I = +$, $e^I = 0$ es modelo de S .
 - $\mathcal{I} = (\mathbb{N}, I)$ con $R^I = <$, $f^I = +$, $e^I = 0$ no es modelo de S .

32

Consistencia de un conjunto de fórmulas

- Def.: Sea S un conjunto de fórmulas de L .
 - ▶ S es **consistente** si S tiene alguna realización (i.e. existe alguna estructura \mathcal{I} de L y alguna asignación A en \mathcal{I} tales que, para toda $F \in S$, $I_A(F) = 1$).
 - ▶ S es **inconsistente** si S no tiene ninguna realización (i.e. para toda estructura \mathcal{I} de L y toda asignación A en \mathcal{I} , existe alguna $F \in S$, tal que $I_A(F) = 0$).
- Ejemplos:
 - ▶ $S = \{(\forall y)R(x,y), (\forall y)f(x,y) = y\}$ es consistente.
 (\mathcal{I}, A) con $\mathcal{I} = (\mathbb{N}, I), R^I = \leq, f^I = +, A(x) = 0$ es realización de S .
 - ▶ $S = \{P(x) \rightarrow Q(x), (\forall y)P(y), \neg Q(x)\}$ es inconsistente.
- Prop.: Sea S un conjunto de fórmulas *cerradas* de L . Entonces S es consistente syss S tiene algún modelo.

33

Consecuencia lógica

- Def.: Sean F una fórmula de L y S un conjunto de fórmulas de L .
 - ▶ F es **consecuencia lógica** de S si todas las realizaciones de S lo son de F . (i.e. para toda estructura \mathcal{I} de L y toda asignación A en \mathcal{I} , si $\mathcal{I}_A \models S$ entonces $\mathcal{I}_A \models F$).
 Se representa por $S \models F$.
 - ▶ Se escribe $G \models F$ en lugar de $\{G\} \models F$.
 - ▶ Se escribe $G \not\models F$ en lugar de $\{G\} \not\models F$.
- Ejemplos:
 - ▶ $(\forall x)P(x) \models P(y)$
 - ▶ $P(y) \not\models (\forall x)P(x)$
 (\mathcal{I}, A) con $\mathcal{I} = (U, I), U = \{1, 2\}, P^I = \{1\}, A(y) = 1$.
 - ▶ $\{(\forall x)(P(x) \rightarrow Q(x)), P(c)\} \models Q(c)$
 - ▶ $\{(\forall x)(P(x) \rightarrow Q(x)), Q(c)\} \not\models P(c)$
 (\mathcal{I}, A) con $\mathcal{I} = (U, I), U = \{1, 2\}, c^I = 1, P^I = \{2\}, Q^I = \{1, 2\}$.
 - ▶ $\{(\forall x)(P(x) \rightarrow Q(x)), \neg Q(c)\} \models \neg P(c)$
 - ▶ $\{P(c), \neg P(d)\} \models c \neq d$

34

Consecuencia lógica e inconsistencia

- $S \models F$ syss $S \cup \{\neg F\}$ es inconsistente.
 - $S \models F$
 \iff para toda estructura \mathcal{I} de L y toda asignación A en \mathcal{I} , si, para todo $G \in S$, $\mathcal{I}_A(G) = 1$ entonces $\mathcal{I}_A(F) = 1$.
 - \iff para toda estructura \mathcal{I} de L y toda asignación A en \mathcal{I} , si, para todo $G \in S$, $\mathcal{I}_A(G) = 1$ entonces $\mathcal{I}_A(\neg F) = 0$.
 - \iff para toda estructura \mathcal{I} de L y toda asignación A en \mathcal{I} , existe alguna $H \in S \cup \{\neg F\}$ tal que $\mathcal{I}_A(H) = 0$.
 - $\iff S \cup \{\neg F\}$ es inconsistente.
- Sean F una fórmula *cerrada* de L y S un conjunto de fórmulas *cerradas* de L . Entonces, son equivalentes
 - ▶ F es consecuencia lógica de S
 - ▶ todos los modelos de S lo son de F .

35

Equivalencia lógica

- Def.: Sean F y G fórmulas de L . F y G son **equivalentes** si para toda estructura \mathcal{I} de L y toda asignación A en \mathcal{I} , $\mathcal{I}_A(F) = \mathcal{I}_A(G)$. Se representa por $F \equiv G$.
- Ejemplos:
 - ▶ $P(x) \not\equiv P(y)$.
 $\mathcal{I} = (\{1, 2\}, I)$ con $P^I = \{1\}$ y $A(x) = 1, A(y) = 2$.
 - ▶ $(\forall x)P(x) \equiv (\forall y)P(y)$.
 - ▶ $(\forall x)(P(x) \wedge Q(x)) \equiv (\forall x)P(x) \wedge (\forall x)Q(x)$.
 - ▶ $(\exists x)(P(x) \wedge Q(x)) \not\equiv (\exists x)P(x) \wedge (\exists x)Q(x)$.
 $\mathcal{I} = (\{1, 2\}, I)$ con $P^I = \{1\}$ y $Q^I = \{2\}$.
- Propiedades: Sean F y G fórmulas *cerradas* de L .
 - ▶ $F \equiv G$ syss $\models F \leftrightarrow G$.
 - ▶ $F \equiv G$ syss $F \models G$ y $G \models F$.

36

Bibliografía

1. C. Badesa, I. Jané y R. Jansana *Elementos de lógica formal*. (Ariel, 2000) pp. 195–259 y 323–326.
2. M.L. Bonet *Apuntes de LPO*. (Univ. Politécnica de Cataluña, 2003) pp. 17–26.
3. C.–L. Chang y R.C.–T. Lee *Symbolic logic and mechanical theorem proving* (Academic Press, 1973) pp. 26–35.
4. J.L. Fernández, A. Manjarrés y F.J. Díez *Lógica computacional*. (UNED, 2003) pp. 64–87.
5. J.H. Gallier *Logic for computer science (foundations of automatic theorem Proving)* (June 2003) pp. 146–186.
6. M. Huth y M. Ryan *Logic in computer science: modelling and reasoning about systems*. (Cambridge University Press, 2000) pp. 90–109 y 128–140.
7. M. Ojeda e I. Pérez de Guzmán *Lógica para la computación (Vol. 2: Lógica de primer orden)* (Ágora, 1997) pp. 1–37 y 49–51.
8. L. Paulson *Logic and proof* (U. Cambridge, 2002) pp. 22–29.