Ciencias computacionales

Propedeútico: Matemáticas Discretas

Lógica

INAOE

Contenido

Material multimedia recomendado:

- https://www.youtube.com/watch?v=6Q7evuBaP1Q
- https://www.youtube.com/watch?v=r9fJfts3Ktk
- https://www.youtube.com/watch?v=WzonHoz4TIA
- https://www.youtube.com/watch?v=fvO6HSL6up0
- https://www.youtube.com/watch?v=fvO6HSL6up0

1 Introducción

¿Qué es la lógica?

- *La ciencia del razonamiento.* Gracias al razonamiento se resuelve un problema, a través de un proceso que extrae conclusiones a partir de premisas. Este proceso es extremandamente complejo, emotivo, compuesto de un ciclo de prueba-error, "iluminado" por momentos de comprensión o intuición.
- Es el estudio de los métodos y principios que se usan para distinguir el razonamiento bueno (correcto) del malo (incorrecto).
  - ¿Un arte o una ciencia?
    - La práctica llevará al perfeccionamiento
    - Análisis de las falacias
    - Errores frecuentes del razonamiento

En lógica se plantean preguntas como *¿tiene solución el problema?* o *¿Se sigue la conclusión de las permisas que se han afirmado o supuesto?*. Si el problema queda resuelto, si las premisas proporcionan las bases adecuadas para afirmar la verdad de la conclusión, entonces el razonamiento es correcto. De lo contrario es incorrecto.

La distinción entre razonamiento correcto e incorrecto es el problema central con el que trata la lógica

1.1 Términos esenciales

1.1.1 Proposición

Proposición es el contenido de una oración el cual puede ser verdadero o falso. Difieren de las preguntas, órdenes y exclamaciones debido a que éstas no pueden ser verdaderas o falsas.

Dos oraciones pueden emplearse para afirmar la misma proposición, por ejemplo:

- Juan ama a María
- María es amada por Juan

Usamos el término proposición para referirnos al contenido que ambas oraciones afirman.

Una proposición se interpreta en un contexto, por ejemplo

- El presidente actual es priista

Dependiendo del momento, esta oración corresponde a un enunciado verdadero (Enrique Peña) o a un enunciado falso (Vicente Fox). Los términos enunciado y proposición no son exactamente sinónimos.

1.1.2 Inferencia

Inferencia es el proceso por el cual se llega a una proposición y se afirma sobre ella en base de una o más proposiciones aceptadas como punto inicial del proceso.

1.1.3 Argumento

En correspondencia a cada inferencia existe un **argumento**. Un argumento es cualquier conjunto de proposiciones de las cuales se dice que una se sigue de las otras, que pretenden apoyar o fundamentar su verdad.

1.1.4 Premisa-conclusión

Un argumento tiene una estructura: **premisa-conclusión**

La **conclusión** de un argumento es la proposición que se afirma con base en las otras proposiciones del argumento.

Las otras proposiciones afirmadas o supuestas para aceptar la conclusión son las **premisas** del argumento.

Ejemplo 1

> Como las sensaciones son esencialmente privadas, no podemos saber cómo es el mundo para otras personas.

Ejemplo 2

> Una superficie gris se ve roja si antes hemos visto una azul verdosa; una hoja de papel se siente muy suave si hemos tocado antes una lija, o rugosa si antes hemos tocado una suave superficie de cristal; el agua de la llave sabe dulce si hemos comido antes alcachofas. Por tanto, una parte de lo que llamamos rojo, suave o dulce debe estar en los ojos, los dedos o la lengua del que ve, toca o prueba.

Ejemplo 3

> Enfriar los átomos equivale a retardar su movimiento, puesto que la temperatura es una medida de qué tan rápido se están moviendo los átomos o las moléculas (el cero absoluto es la inmovilidad total).

IMPORTANTE:

- ☞ Ninguna proposición por sí misma, en forma aislada, es una premisa o una conclusión.
- ☞ Una premisa solamente aparece como supuesto en un argumento.
- ☞ Una conclusión solamente aparece en un argumento y se fundamenta en las otras proposiciones del argumento.
- ☞ Son términos relativos. Como ”empleador” y ”empleado”: depende de un contexto.
- ☞ El orden no es relevante.

Ejemplo 4

> Puesto que la libertad y el bienestar son las condiciones necesarias de la acción y en general de la acción exitosa, cada agente debe reconocer estas condiciones como bienes necesarios para sí mismo, puesto que sin ellas no sería capaz de actuar para conseguir un propósito determinado, sea en absoluto o con las oportunidades generales de lograr el éxito.

2 Sintáxis y semática de la lógica proposicional

2.1 Fundamentos de lógica

En el desarrollo de cualquier teoría matemática se hacen afirmaciones en forma de frases. Tales afirmaciones, verbales o escritas, denominadas **enunciados** o **proposiciones**, son frases declarativas verdaderas o falsas.

Las proposiciones se representan con letras minpusculas:

p: Matemáticas Discretas es un curso propedeúico de la Maestría en Ciencias Computacionales

q: Juan Rulfo escribió "Pedro Páramo"

Las proposiciones más sencillas posibles se denominan **atómicas**.

Definición

Las proposiciones constituidas por proposiciones atómicas y otras partículas que sirven de nexos se llaman **compuestas**.

2.1.1 Conectivas básicas

Estas proposiciones pueden considerarse como *primitivas*, ya que en realidad no se pueden descomponer en partes más simples. Las proposiciones primitivas se utilizan con *conectivos lógicos* para formar enunciados compuestos.

Definición

Un **conectivo lógico** es una partícula que se utiliza para formar las proposiciones compuestas, es decir, es un elemento de lenguaje que permite construir proposiciones nuevas a partir de las existentes, obteniendo así nuevos significados.

Las conectivas lógicas son:

$\neg$	negación
$\wedge$	conjunción
$\vee$	disyunción
$\veebar$	disyunción mutuamente exclusiva
$\rightarrow$	implicación
$\leftrightarrow$	doble implicación

- **Negación.** La negación de una proposición p se denota se denota por $\overline{p}$ o $\neg p$, y se lee "no p".

Ejemplo 1

Para p : Combinatoria es un curso obligatorio
$\neg p$ es la proposición "Combinatoria **no** es un curso obligatorio".

- **Conjunción.** la conjunción de p, q se denota por $p \wedge q$, y se lee "p y q".

Ejemplo 2

Para
∗ p : Combinatoria es un curso obligatorio
∗ q : Estudio la especialidad de computación

Se lee:
"Combinatoria es un curso obligatorio **y** yo estudio la especialidad de computación".

- **Disyunción.** La expresión $p \vee q$ denota la disyunción de p, q, y se lee "p o q".

Ejemplo 3

Para
- p : Combinatoria es un curso obligatorio
- q : Estudio la especialidad de computación

Se lee:
"Combinatoria es un curso obligatorio **o** yo estudio la especialidad de computación".

Aquí se utiliza la conjunción "o" en sentido *inclusivo.* En consecuencia, $p \vee q$ es verdadera si una o la otra, o ambas proposiciones p, q son verdaderas. Esto suele indicase escribiendo "y/o".

- **Disyunción mutuamente exclusiva.** El "o" exclusivo se denota mediante $p \veebar q$. La proposición compuesta $p \veebar q$ es verdadera si una u otra, pero no ambas proposiciones p, q son verdaderas.

Ejemplo 4

Para
- p : Combinatoria es un curso obligatorio
- q : Estudio la especialidad de computación

Se lee:
"Combinatoria es un curso obligatorio **o** yo estudio la especialidad de computación, **pero no ambas**".

- **Implicación.** Se dice que "p implica q" y se escribe $p \rightarrow q$ para designar la implicación de q por p. Opcionalmente, se puede decir:

  a) si p, entonces q
  b) p es suficiente para q
  c) p sólo si q
  d) q es necesario para p

  Una traducción verbal de $p \rightarrow q$ para el ejemplo es: "Si combinatoria es un curso obligatorio, **entonces** yo estudio la especialidad de computación".

  La proposición p se denomina *hipótesis* de la implicación y q se denomina *conclusión.*

- **Doble implicación.** La doble implicación de dos preposiciones p, q se denota por $p \leftrightarrow q$, y se lee:

  a) p es equivalente a q
  b) p si, y sólo si q
  c) es necesario y suficiente para q

  Para las p, q anteriores, "Combinatoria es un curso obligatorio **si, y sólo si**, yo estudio la especialidad de computación" expresa el significado de $p \leftrightarrow q$.

En las proposiciones de la lógica proposicional pueden incluirse signos que agrupen proposiciones, por ejemplo:

$\neg p \vee (q \leftrightarrow r)$

$\neg(p \rightarrow r) \wedge (q \leftrightarrow r)$

$(p \wedge q) \rightarrow (q \vee \neg r)$

2.1.2 Tablas de verdad

Los **valores de verdad** de las proposiciones compuestas pueden ser descritos por tablas de verdad. Las tablas de verdad expresan las circunstancias en las que una proposición de la lógica proposicional es *verdadera* (1) o *falsa* (0):

Definición

Una tabla de verdad de una proposición presenta los valores de verdad de la proposición para todas las asignaciones posibles de la proposiciones atómicas

p	$\neg p$
0	1
1	0

p	q	$p \wedge q$	$p \vee q$	$p \veebar q$	$p \rightarrow q$	$p \leftrightarrow q$
0	0	0	0	0	1	1
0	1	0	1	1	1	0
1	0	0	1	1	0	0
1	1	1	1	0	1	1

- Una conjunción es verdadera cuando p y q, ambas, son verdaderas.

p	q	$p \wedge q$
1	1	1
1	0	0
0	1	0
0	0	0

- Una disyunción exclusiva es verdadera cuando p es verdadera y q falsa o bien cuando p es falsa y q verdadera.

p	q	$p \veebar q$
1	1	0
1	0	1
0	1	1
0	0	0

- Una implicación es verdadera cuando p y q son verdaderas o cuando p es falsa.

p	q	$p \rightarrow q$
1	1	1
1	0	0
0	1	1
0	0	1

2.1.3 Tautología y contradicción

Definición

Una proposición de la lógica proposicional es una **tautología**, denotada como T_0 , si es verdadera para todas las asignaciones de verdad de sus proposiciones componentes.

p	q	$p \to p \vee q$
1	1	1
1	0	1
0	1	1
0	0	1

Si para las proposiciones s_1, s_2, resulta que $s1 \Leftrightarrow s2$ es una tautología, entonces s_1 y s_2 deben tener la misma tabla de verdad, de modo que $s_1 \Leftrightarrow s_2$.

Definición

Una proposición de la lógica proposicional es una **contradicción**, denotada como F_0 , si es falsa para todas las asignaciones de verdad de sus proposiciones componentes.

p	$p \wedge (\neg p)$
1	0
0	1

2.1.4 Precedencia de los operadores

En proposiciones compuestas como $r \vee p \to q$, ¿cuál es el orden de la aplicación de los operadores lógicos?.

Se usan parentesis para especificar el orden:

$r \vee (p \to q)$

Si no se emplean paréntensis, se emplea la lista de precedencia

Operador	**Precedencia**
$\neg$	1
$\wedge$	2
$\vee$	3
$\to$	4
$\leftrightarrow$	5

2.1.5 Equivalencia lógica

Definición

Dos proposiciones s_1 y s_2 son lógicamente **equivalentes**, denotado como $s_1 \Leftrightarrow s_2$, si y sólo si la proposición s_1 es verdadera cuando s_2 es verdadera, y s_1 es falsa cuando s_2 es falsa.

Dos proposiciones son equivalentes lógicamente si y solo si sus valores de verdad (tablas de verdad) son idénticos.

Ejemplo 5

Probar que $q \wedge (\neg r \to p) \Leftrightarrow (r \wedge q) \vee (p \wedge q)$

p	q	r	$\neg r$	$\neg r \to p$	$q \wedge (\neg r \to p)$	$r \wedge q$	$p \wedge q$	$(r \wedge q) \vee (p \wedge q)$
F	F	F	T	F	**F**	F	F	**F**
F	F	T	F	T	**F**	F	F	**F**
F	T	F	T	F	**F**	F	F	**F**
F	T	T	F	T	**T**	T	F	**T**
T	F	F	T	T	**F**	F	F	**F**
T	F	T	F	T	**F**	F	F	**F**
T	T	F	T	T	**T**	F	T	**T**
T	T	T	F	T	**T**	T	T	**T**

Table 1: Donde F = *false* y T = *true*

De acuerdo con la tabla de verdad $q \wedge (\neg r \to p)$ y $(r \wedge q) \vee (p \wedge q)$ son proposiciones equivalentes.

2.1.6 Leyes de equivalencia

Otra forma de com probar si dos proposiciones son lógicamente equivalentes es aplicando las leyes de la lógica. En general, las leyes de la lógica son útiles para simplificar una proposición.

Para cualesquiera proposiciones lógicas p, q y r:

$\neg\neg p \Leftrightarrow p$	Doble negación
$\neg(p \vee q) \Leftrightarrow \neg p \wedge \neg q$	DeMorgan
$\neg(p \wedge q) \Leftrightarrow \neg p \vee \neg q$	
$p \vee q \Leftrightarrow \vee p$	Conmutativas
$p \wedge q \Leftrightarrow q \wedge p$	
$p \vee (q \vee r) \Leftrightarrow (p \vee q) \vee r$	Asociativas
$p \wedge (q \wedge r) \Leftrightarrow (p \wedge q) \wedge r$	
$p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r)$	Distributivas
$p \wedge (q \vee r) \Leftrightarrow (p \wedge q) \vee (p \wedge r)$	
$p \vee p \Leftrightarrow p$	Idempotentes
$p \wedge p \Leftrightarrow p$	
$p \vee F_0 \Leftrightarrow p$	Identidad
$p \wedge T_0 \Leftrightarrow p$	
$p \vee \neg p \Leftrightarrow T_0$	Inversas
$p \wedge \neg p \Leftrightarrow F_0$	
$p \vee T_0 \Leftrightarrow T_0$	Dominación
$p \wedge F_0 \Leftrightarrow F_0$	
$p \vee (p \wedge q) \Leftrightarrow p$	Absorción
$p \wedge (p \vee q) \Leftrightarrow p$	

Definición

Sea s una proposición de la lógica proposicional. Si s sólo contiene conectivas lógicas conjuntivas y disyuntivas, entonces el **dual** de s, denotado como s^d, es la proposición que se obtiene al reemplazar cada ocurrencia de $\wedge$ y $\vee$ por $\vee$ y $\wedge$, respectivamente, y cada ocurrencia de T_0 y F_0 por F_0 y T_0, respectivamente.

Ejemplo 6

Los siguientes pares de proposiciones son duales.

a) $s_1 : p \vee T_0$, $s_1^d : p \wedge F_0$

b) $s_2 : (p \wedge \neg q) \vee (r \wedge T_0)$, $s_2^d : (p \vee \neg q) \wedge (r \vee F_0)$

Definición

Teoremoa del principio de la dualidad.

Sean s y t proposiciones de la lógica proposicional, si $s \Leftrightarrow t$ entonces $s^d \Leftrightarrow t^d$

Analicem os los valores de verdad de las proposiciones $p \to q$, $\neg q \to \neg p$, $q \to p$ y $\neg p \to \neg q$

p	q	$p \to q$	$\neg q \to \neg p$	$q \to p$	$\neg p \to \neg q$
0	0	1	1	1	1
0	1	1	1	0	0
1	0	0	0	1	1
1	1	1	1	1	1

De acuerdo a la tabla previa, podemos afirmar que $(p \to q) \Leftrightarrow (\neg q \to \neg p)$ y $(q \to p) \Leftrightarrow (\neg p \to \neg q)$. La proposición $\neg q \to \neg p$ se conoce como la **contrapositiva** de la implicación $p \to q$. La proposición $q \to p$ se denomina la **recíproca** de $p \to q$ y $\neg q \to \neg p$ se conoce como la inversa de $p \to q$. Nótese que

$$(p \to q) \not\Leftrightarrow (q \to p) \text{ y}$$
$$(\neg p \to \neg q) \not\Leftrightarrow (\neg q \to \neg p)$$

Ejemplo 7

Se tiene en 1a una red con un interruptor, 1b y 1c tienen dos interruptores (independientes) cada una. En la red de 1b, la corriente fluye de T_1 a T_2 si cualquiera de los dos interruptores p, q está cerrado. Esta red se denomina paralela y se representa por $p \vee q$. La red 1c necesita que los dos interruptores p, q estén cerrados para que la corriente fluya de T_1 a T_2. Aquí, los interruptores están en serie y esta red se representa por $p \wedge q$.

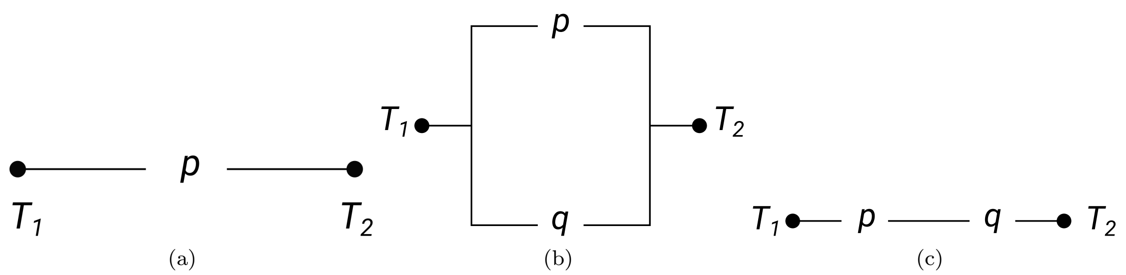


Figura 1: Red 1

Ejemplo 8

Suponga que tenemos la red ilustrada en la Figura 2, en la cual los interruptores no son independientes entre sí. Esta red puede representarse como la proposición lógica:

$$(p \vee q \vee r) \wedge (p \vee t \vee \neg q) \wedge (p \vee \neg t \vee r)$$

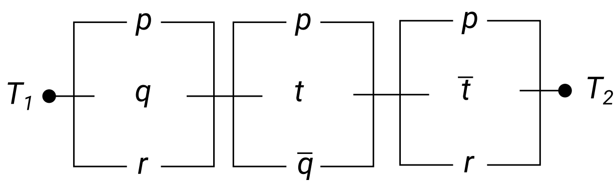


Figura 2: Red 2

y suponga también que deseamos simplificar esta red. Para simplificar la expresión, debemos buscar la **equivalencia lógica** de la proposición original y proposiciones más simples.

	Simplificación	Razones
	$(p \vee q \vee r) \wedge (p \vee t \vee \neg q) \wedge (p \vee \neg t \vee r)$	
$\Leftrightarrow$	$p \vee [(q \vee r) \wedge (t \vee \neg q) \wedge (\neg t \vee r)]$	Ley distributiva
$\Leftrightarrow$	$p \vee [((q \vee r) \wedge t) \vee ((q \vee r) \wedge \neg q)) \wedge (\neg t \vee r)]$	Ley distributiva
$\Leftrightarrow$	$p \vee [(((q \vee r) \wedge t) \vee ((q \wedge \neg q) \vee (r \wedge \neg q))) \wedge (\neg t \vee r)]$	Ley distributiva
$\Leftrightarrow$	$p \vee [(((q \vee r) \wedge t) \vee (F_0 \vee (r \wedge \rightarrow q))) \wedge (\neg t \vee r)]$	Ley inversa
$\Leftrightarrow$	$p \vee [(((q \vee r) \wedge t) \vee (r \wedge \neg q)) \wedge (\neg t \vee r)]$	Ley del elemento neutro
$\Leftrightarrow$	$p \vee [(((q \vee r) \wedge t) \wedge (\neg t \vee r)) \vee ((r \wedge \neg q) \wedge (\neg t \vee r))]$	Ley distributiva
$\Leftrightarrow$	$p \vee [((q \vee r) \wedge (t \wedge (\neg t \vee r))) \vee ((r \wedge \neg q) \wedge (\neg t \vee r))]$	Ley asociativa
$\Leftrightarrow$	$p \vee [((q \vee r) \wedge ((t \wedge \neg t) \vee (t \wedge r))) \vee ((r \wedge q) \wedge (\neg t \vee r))]$	Ley distributiva
$\Leftrightarrow$	$p \vee [((q \vee r) \wedge (F_0 \vee (t \wedge r))) \vee ((r \wedge \neg q) \wedge (\neg t \vee r))]$	Ley inversa
$\Leftrightarrow$	$p \vee [((q \vee r) \wedge (t \wedge r)) \vee ((r \wedge \neg q) \wedge (\neg t \vee r))]$	Ley del elemento neutro
$\Leftrightarrow$	$p \vee [((q \vee r) \wedge (t \wedge r)) \vee (((r \wedge \neg q) \wedge \neg t) \vee ((r \wedge \neg q) \wedge r))]$	Ley distributiva
$\Leftrightarrow$	$p \vee [((q \vee r) \wedge (t \wedge r)) \vee ((r \wedge \neg q \wedge \neg t) \vee (r \wedge \neg q \wedge r))]$	Ley asociativa
$\Leftrightarrow$	$p \vee [((q \vee r) \wedge (t \wedge r)) \vee ((r \wedge \neg q \wedge \neg t) \vee (r \wedge \neg q))]$	Ley idempotente
$\Leftrightarrow$	$p \vee [((q \vee r) \wedge (t \wedge r)) \vee (r \wedge \neg q)]$	Ley absorción
$\Leftrightarrow$	$p \vee [((q \wedge (t \wedge r)) \vee (r \wedge (t \wedge r))) \vee (r \wedge \neg q)]$	Ley distributiva
$\Leftrightarrow$	$p \vee [((q \wedge (t \wedge r)) \vee (t \wedge r)) \vee (r \wedge \neg q)]$	Leyes asociativa e idempotente
$\Leftrightarrow$	$p \vee [(t \wedge r) \vee (r \wedge \neg q)]$	Ley de absorción
$\Leftrightarrow$	$p \vee [(r \wedge t) \vee (r \wedge \neg q)]$	Ley conmutativa
$\Leftrightarrow$	$p \vee [r \wedge (t \vee \neg q)]$	Ley distributiva

Así $(p \vee q \vee r) \wedge (p \vee t \vee \neg q) \wedge (p \vee \neg t \vee r) \Leftrightarrow p \vee [r \wedge (t \vee \neg q)]$, y la red de la Figura 3 es equivalente a la red original, pero esta red tiene 4 interruptores, 5 menos que la original.

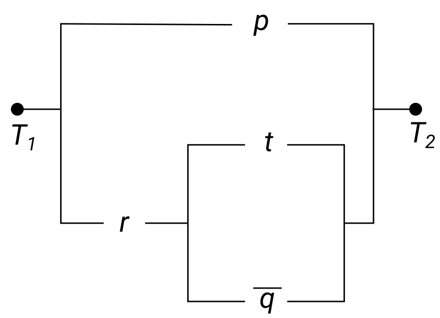


Figura 3: Red simplificada

2.1.7 Reglas de inferencia

Una demostración es un argumento correcto (bien razonado, lógicamente válido) y completo (claro, detallado) que rigurosamente establece la veracidad de una proposición matemática.

Una **regla de inferencia** es un patrón que establece que si se conocen un conjunto de proposiciones antecedentes verdaderas, entonces se puede deducir que cierta proposición consecuente relacionada es verdadera. Es decir, permiten determinar la validez de un argumento para demostrar un teorema.

$$(p_1 \wedge p_2 \wedge p_3 \wedge \cdots \wedge p_n) \to q$$

$p_1, p_2, p_3, \ldots, p_n$ se conocen como las **premisas** del argumento, y q se conoce como la **conclusión** del argumento.

Una manera de inferir la conclusión es dem ostrar que la proposición correspondiente a las premisas es una tautología.

Definición

Si p y q son proposiciones arbitrarias, y $p \to q$ es una tautología, decimos que p **implica lógicamente** a q y escribimos $p \Rightarrow q$.

Ejemplo 9

Sean $p_1 : p$, $p_2 : (p \wedge r) \to s$ y $q : r \to s$ Verifique con una tabla de verdad que $(p_1 \wedge p_2)$ implica lógicamente a q. Es decir, tenemos que demostrar $(p_1 \wedge p_2) \to q$ es una tautología.

p_1				p_2	q	$(p_1 \wedge p_2)$	$(p_1 \wedge p_2) \to q$
p	r	s	$p \wedge r$	$(p \wedge r) \to s$	$r \to s$	$[p \wedge ((p \wedge r) \to s)]]$	$[p \wedge ((p \wedge r) \to s)]] \to (r \to s)$
0	0	0	0	1	1	0	1
0	0	1	0	1	1	0	1
0	1	0	0	1	0	0	1
0	1	1	0	1	1	0	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	1	0	0	0	1
1	1	1	1	1	1	1	1

Las reglas de inferencia postulan asociaciones conocidas y probadas de implicación lógica. Con las tablas de verdad correspondientes, que son tautologías, puede dem ostrarse que estas reglas son implicaciones lógicas.

p	Regla de adición	$\therefore p \vee q$
$p \wedge q$	Regla de simplificación	$\therefore p$
p	Regla de conjunción	$q \therefore p \wedge q$

- **Modus Ponens**

$$\begin{array}{l|l} p & \text{premisas} \\ p \to q & \\ \hline \therefore q & \text{conclusión} \end{array} \tag{1}$$

- **Ley del silogismo**

$$\begin{array}{c} p \to q \\ q \to r \\ \hline \therefore p \to r \end{array} \tag{2}$$

- **Modus Tollens**

$$\begin{array}{c} p \to q \\ \neg q \\ \hline \therefore \neg p \end{array} \tag{3}$$

- **Regla de la conjunción**

$$\begin{array}{l} p \\ q \\ \hline \therefore p \wedge q \end{array} \tag{4}$$

- **Regla del silogismo disyuntivo**

$$\begin{array}{l} p \vee q \\ \neg p \\ \hline \therefore q \end{array} \tag{5}$$

- **Regla de la contradicción**

$$\begin{array}{l} \neg p \to F_0 \\ \hline \therefore p \end{array} \tag{6}$$

- **Regla de simplificación conjuntiva**

$$\begin{array}{l} p \wedge q \\ \hline \therefore p \end{array} \tag{7}$$

- **Regla de simplificación disyuntiva**

$$\begin{array}{l} p \\ \hline \therefore p \vee q \end{array} \tag{8}$$

- **Regla de demostración condicional**

$$\begin{array}{l} p \wedge q \\ p \to (q \to r) \\ \hline \therefore r \end{array} \tag{9}$$

- **Regla de demostración por casos**

$$\begin{array}{l} p \to r \\ q \to r \\ \hline \therefore (p \vee q) \to r \end{array} \tag{10}$$

- **Regla del dilema constructivo**

$$\begin{array}{l} p \to q \\ r \to s \\ p \vee r \\ \hline \therefore q \vee s \end{array} \tag{11}$$

- **Regla del dilema destructivo**

$$\begin{array}{l} p \to q \\ r \to s \\ \neg q \vee \neg s \\ \hline \therefore \neg p \vee \neg r \end{array} \tag{12}$$

Ejemplo 10

Aplique las reglas de inferencia *modus ponens*, *modus tollens* y *regla del silogismo* para demostrar la validez del siguiente argumento.

$$\begin{array}{l} p \to r \\ r \to s \\ t \vee \neg s \\ \neg t \vee u \\ \neg u \\ \hline \therefore \neg p \end{array}$$

Pasos	**Justificación**
1) $p \to r, r \to s$	Premisas
2) $p \to s$	1) + Ley del silogismo
3) $t \vee \neg s$	Premisa
4) $\neg s \vee t$	3) + Comnutativa de disyunción
5) $s \to t$	4) + $\neg s \vee t \Rightarrow s \to t$
6) $p \to t$	2) + 5) + Ley del silogismo
7) $\neg t \vee u$	Premisa
8) $t \to u$	7) + $\neg t \vee u \Rightarrow t \to u$
9) $p \to u$	6) + 8) + Ley del silogismo
10) $\neg u$	Premisa
11) $\therefore \neg p$	9) + 10) + modus tollens

2.1.8 Cuantificadores

Considere las siguiente afirmaciones:

- Todos los gatos tienen cola.
- A algunas personas les gusta la carne cruda.
- Todos conseguimos un descanso de vez en cuando.

Todas estas afirmaciones indican qué tan frecuentemente ciertas cosas son verdaderas. En cálculo de predicados se usan **cuantificadores** en este contexto.

Definición

Sea A una expresión, y sea x una variable. Si queremos indicar que A es verdadera para **todos los valores posibles** de x, escribimos $\forall x A$. Aquí $\forall x$ es llamado el **cuantificador universal**, y A se llama el **ámbito** o **alcance del cuantificador**. La variable x se dice que está ligada al cuantificador. El símbolo $\forall$ se pronuncia "para todo".

El cuantificador y la variable ligada que la sigue tienen que ser tratados como una unidad, y esta unidad actúa como una conectiva unaria. Oraciones que contienen palabras como *cada, cada uno*, y *todos* usualmente indican **cuantificación universal**. Tales oraciones deben ser reescritas de tal forma que empiecen con "para cada x", lo cual se traduce como $\forall x$.

Ejemplo 11

Exprese "*Todos conseguimos un descanso de vez en cuando*" en cálculo de predicados.

Definimos B de tal forma que signifique "*conseguir un descanso de vez en cuando*". Por tanto $B(x)$ significa que x consigue un descanso de vez en cuando. La palabra "**Todos**" indica que esto es verdadero para toda x. Esto lleva a la traducción $\forall x B(x)$.

Definición

Si A representa una expresión, y x representa a una variable. Si queremos indicar que A es verdadero para **al menos un valor de** x, escribimos $\exists x\ A$. Esto se pronuncia "existe una x tal que A". Aquí $\exists x$ es llamado el **cuantificador existencial**, y A se llama el alcance o ámbito del cuantificador existencial. La variable x se dice que está ligada al cuantificador.

Afirmaciones que contienen frases como *algunos, al menos uno* sugieren cuantificación existencial. Y deben ser reescritos como "Hay un x tal que", lo cual se traduce en $\exists x$.

Ejemplo 12

Sea P la propiedad "*le gusta la carne cruda*". Entonces $\exists x\ P(x)$ se puede traducir como "*Hay personas a las que les gusta la carne cruda*" o "*A algunas personas les gusta la carne cruda*".

Los cuantificadores se pueden anidar, como se muestra en los siguientes ejemplos:

Ejemplo 13

Traducir la oración "Hay alguien que conoce a todos" en lenguaje de cálculo de predicados.

Para hacer esto, use $C(x, y)$ para expresar el hecho de que x conoce a y. Escribimos informalmente $\exists x$(x conoce a todos). Aquí "x conoce a todos" está todavía en español y significa que **para todo** y **es verdad que** x **conoce a** y. Por tanto x conoce a todos $= \forall y C(x, y)$. Finalmente se añade el cuantificador existencial y obtenemos $\exists x \forall y C(x, y)$.

Ejemplo 14

Traducir "Cualquiera tiene alguien quien es su madre" en cálculo de predicados.

Definimos M como el predicado "madre"; esto es, $M(x, y)$ significa que "x es la madre de y". La oración "Alguien es la madre de y" se traduce como $\exists x M(x, y)$. Para expresar que esto debe ser verdad para toda y, agregamos el cuantificador universal, lo cual queda como $\forall y \exists x M(x, y)$

La oración "Nadie es perfecto" también incluye un cuantificador, "Nadie", el cual es la ausencia de un individuo con una cierta propiedad. En cálculo de predicados el hecho de que nadie tenga una propiedad P, no puede ser expresada directamente. Para expresar el hecho de que no hay x para la cual una expresión A es verdadera, uno puede usar $\neg\exists x A$ o $\forall x \neg P(x)$, indica que nadie es perfecto. En el primer caso, decimos, en traducción verbal, "No se da el caso de que exista alguien quien sea perfecto", mientras que en el segundo caso decimos "Para cualquiera, no se da el caso de que sea perfecto". Los dos métodos para expresar que nadie es A deben por supuesto ser lógicamente equivalentes; esto es:

$$\neg\exists x A \equiv \forall x \neg P(x)$$

2.2 Sintáxis de la lógica proposicional

El alfabeto proposicional esta conformado por:

- variables proposicionales: $p_0, p_1, \ldots; p, q, r$
- conectivas lógicas:
  - monaria: $\neg$ (negación)
  - binarias: $\wedge$ (conjunción), $\vee$ (disyunción), $\rightarrow$ (condicional), $\leftrightarrow$ (bicondicional)
- símbolos auxiliares: "(" y ")"

Las variables proposicionales son fórmulas (**fórmulas atómicas**). Si F y G son fórmulas entonces también lo son:

$$\neg F, (F \wedge G), (F \vee G), (F \rightarrow G) \text{ y } (F \leftrightarrow G)$$

Ejemplo 1

- Fórmulas: p, $(p \vee \neg q)$, $\neg(p \vee p)$, $((p \rightarrow q) \vee (q \rightarrow p))$
- No fórmulas: (p), $p \vee \neg p$, $(p \vee \wedge q)$

2.2.1 Fórmulas proposicionales (BNF)

Notaciones:

- $p, q, r, \ldots$ representarán variables proposicionales.
- $F, G, H, \ldots$ representarán fórmulas.
- VP representa el conjunto de los variables proposicionales.
- $Prop$ representa el conjunto de las fórmulas.
- * representa una conectiva binaria.

La forma de Backus Naur (BNF) de las fórmulas proposicionales es:

$$F ::= p|\neg G|(F \wedge G)|(F \vee G)|(F \rightarrow G)|(F \leftrightarrow G)$$

2.2.2 Definiciones por recursión sobre fórmulas

El número de paréntesis de una fórmula F se defina recursivamente por:

$$np(F) = \begin{cases} 0, & \text{si F es atómica;} \\ np(G), & \textit{si} \text{ F es } \neg G; \\ 2 + np(G) + np(H), & \text{si F es } (G * H) \end{cases}$$

Ejemplo 2

El número de paréntesis de las siguientes proposiciones es:

- $np(p) = 0$
- $np(q) = 0$
- $np(\neg q) = 0$
- $np((\neg q \vee p)) = 2$
- $np((p \rightarrow (\neg q \vee p))) = 4$

2.2.3 Demostración por inducción sobre fórmulas

Sea P una propiedad sobre las fórmulas que verifica las siguientes condiciones:

- Todas las fórmulas atómicas tiene la propiedad P.
- Si F y G tienen la propiedad P.
- Si F y G tiene la propiedad P, entonces $\neg F$, $(F \wedge G)$, $(F \vee G)$, $(F \rightarrow G)$ y $(F \leftrightarrow G)$, tienen la propiedad P

Entonces todas las fórmulas proposicionales tienen la propiedad P.

☞ ***Propiedad***. Todas las fórmulas proposicionales tienen un número par de paréntesis.

2.2.4 Criterio de reducción de paréntesis

Algunas forma de reducir los paréntesis son:

- Eliminar los paréntesis externos
  - $F \wedge G$ es una abreviatura de $(F \wedge G)$
- Por precedencia de asociacón de conectivas: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$.
  - $F \wedge G \rightarrow \neg F \vee G$ es una abreviatura de $((F \wedge G) \rightarrow (\neg F \vee G))$.
- Si una conectiva se usa repetidamenente, se asocia por la derecha

$$\begin{array}{ccc} F \vee G \vee H & \text{abrevia} & (F \vee (G \vee H)) \\ F \wedge G \wedge H \rightarrow \neg F \vee G & \text{abrevia} & ((F \wedge (G \wedge H)) \rightarrow (\neg F \vee G)) \end{array}$$

2.2.5 Subfórmulas

Definición

El conjunto ***Subf(F)*** de las **subfórmulas** de una fórmula F se define recursivamente por:

$$subf(F) = \begin{cases} \{F\}, & \text{si F es atómica;} \\ \{F\} \cup Subf(G), & \textit{si} \text{ F es } \neg G; \\ \{F\} \cup Subf(G) \cup Subf(H), & \text{si F es } (G * H) \end{cases}$$

Ejemplo 3

- $Subf(p) = \{p\}$
- $Subf(q) = \{q\}$
- $Subf(\neg q) = \{\neg q, q\}$
- $Subf(\neg q \vee p) = \{\neg q \vee r, \neg q, q, p\}$
- $Subf(p \rightarrow \neg q \vee p) = \{p \rightarrow \neg q \vee p, p, \neg q \vee p, \neg q, q\}$

2.3 Sémantica proposicional

2.3.1 Valores y funciones de verdad

Los valores de verdad ($\mathbb{B}$) son **1** es para verdadero y **0** es para falso. Así, las funciones de verdad son las siguientes:

$$H_{\neg} : \{0,1\} \rightarrow \{0,1\} \text{ t.q (tal que) } H_{\neg}(i) = \begin{cases} 1, & \text{si } i = 0; \\ 0, & \text{si } i = 1 \end{cases}$$

$$H_{\wedge} : \{0,1\}^2 \rightarrow \{0,1\} \text{ t.q } H_{\wedge}(i,j) = \begin{cases} 1, & \text{si } i = j = 1; \\ 0, & \text{en otro caso.} \end{cases}$$

$$H_{\vee} : \{0,1\}^2 \rightarrow \{0,1\} \text{ t.q } H_{\vee}(i,j) = \begin{cases} 0, & \text{si } i = j = 0; \\ 1, & \text{en otro caso.} \end{cases}$$

$$H_{\to} : \{0,1\}^2 \to \{0,1\} \text{ t.q } H_{\to}(i,j) = \begin{cases} 0, & \text{si } i = 1, j = 0; \\ 1, & \text{en otro caso.} \end{cases}$$

$$H_{\leftrightarrow} : \{0,1\}^2 \leftrightarrow \{0,1\} \text{ t.q } H_{\leftrightarrow}(i,j) = \begin{cases} 1, & \text{si } i = j; \\ 0, & \text{en otro caso.} \end{cases}$$

Las funciones de verdad también se representan mediante tablas de verdad (ver sub-sección 2.1.2).

Definición

Una **interpretación** es una aplicación $I : VP \to \mathbb{B}$.

Para cada interpretación I existe una única aplicación $I' : Prop \to \mathbb{B}$ tal que:

$$I'(F) = \begin{cases} I(F), & \text{si F es atómica;} \\ H_{\neg}(I'(G)), & \text{si} F = \neg G; \\ H_{\neg}(I'(G), I'(H)), & \text{si } F = G * H \end{cases}$$

Se dice que $I'(F)$ es el **valor de verdad de F respecto de I**.

Ejemplo 1

Sea $F = (p \vee q) \wedge (\neg q \vee r)$.

– Donde el valor de F en una interpretación I_1 tal que $I_1(p) = I_1(r) = 1, I_1(q) = 0$.
Sustituyendo los valores, la interpretación queda de la siguiente manera:

$$\begin{array}{ccc} (p \vee q) & \wedge & (\neg q \vee r) \\ (1 \vee 0) & \wedge & (\neg 0 \vee 1) \\ 1 & \wedge & (1 \vee 1) \\ 1 & \wedge & 1 \\ & 1 & \end{array}$$

– Donde el valor de F en una interpretación I_2 tal que $I_2(r) = 1, I_2(p) = I_2(q) = 0$
Sustituyendo los valores, la interpretación queda de la siguiente manera:

$$\begin{array}{ccc} (p \vee q) & \wedge & (\neg q \vee r) \\ (0 \vee 0) & \wedge & (\neg 0 \vee 1) \\ 0 & \wedge & (1 \vee 1) \\ 0 & \wedge & 1 \\ & 0 & \end{array}$$

☞ ***Propiedad***. Sea F una fórmula y I_1, I_2 dos interpretaciones. Si $I_1(p) = I_2(p)$ para todas las variables proposicionales de F, entonces $I_1'(F) = I_2'(F)$.

2.3.2 Modelos y satisfacibilidad

Definición

I es **modelo** F si $I(F) = 1$.

$$I \models F$$

Ejemplo 2

continuando con el ejemplo anterior (1):

- Si $I_1(p) = I_1(r) = 1, I_1(q) = 0$, entonces $I_1 \models (p \vee q) \wedge (\neg q \vee r)$.
- Si $I_2(r) = 1, I_2(p) = I_2(q) = 0$, entonces $I_2 \not\models (p \vee q) \wedge (\neg q \vee r)$.

Definición

F es **satisfacible** si F tiene algún modelo.

Ejemplo 3

$(p \to q) \wedge (q \to r)$ es satisfacible para $I(p) = I(q) = I(r) = 0$

$$\begin{array}{ccc} (p \to q) & \wedge & (q \to r) \\ (0 \to 0) & \wedge & (0 \to 0) \\ 1 & \wedge & 1 \\ & 1 & \end{array}$$

Definición

F es **insatisfacible** si F **NO** tiene algún modelo.

Ejemplo 4

$p \wedge \neg p$ es insatisfacible, debido a que no hay un modelo para cualquier valor de I.

p	$\neg p$	$p \wedge \neg p$
1	0	0
0	1	0

2.3.3 Tautologías y contradicciones

Definición

F es una **tautología** (o **válida**) si **TODA** interpretación es modelo de F.

$$\models F$$

Definición

F es una **contradicción** si **NINGUNA** interpretación es modelo de F

$$\models F$$

Definición

F es **contingente** si no es tautología ni contradicción

$$\models F$$

Ejemplo 5

De acuerdo con la siguiente tabla de verdad:

p	q	$p \to q$	$q \to p$	$(p \to q) \vee (q \to p)$	$\neg(p \to q)$	$(p \to q) \wedge \neg(p \to q)$
1	1	1	1	1	0	0
1	0	0	1	1	1	0
0	1	1	0	1	0	0
0	0	1	1	1	0	0

1. $(p \to q) \vee (q \to p)$ es una tautología
2. $(p \to q) \wedge \neg(p \to q)$ es una contradicción
3. $p \to q$ es contingente

La clasificación de las fórmulas es la siguiente:

<table>
<tr><td colspan="3">Todas las fórmulas</td></tr>
<tr><td>Tautologías</td><td>Contingentes</td><td>Contradicciones</td></tr>
<tr><td>Verdadera en todas las interpretaciones</td><td>Verdadera en algunas interpretaciones y falsa en otras</td><td>Falsa en todas las interpretaciones</td></tr>
<tr><td colspan="2">Satisfacibles</td><td>Insatisfacibles</td></tr>
<tr><td colspan="3">Todas las fórmulas</td></tr>
</table>

2.3.4 Satisfacibilidad y validéz

Un **problema SAT** es: dada una F determinar si es satisfacible. Mientras que un **problema TAUT** es: dada F determinar si es tautología.

Las siguientes son relaciones entre satisfacibilidad y tautologicidad:

- F es tautología $\Leftrightarrow \neg F$ es insatisfacible.
- F es tautología $\Rightarrow F$ es satisfacible.
- F es satisfacible $\not\Rightarrow \neg F$ es insatisfacible.

Ejemplo 6

$p \to q$ es satisfacible para $I(p) = I(q) = 1$

$\neg(p \to q)$ es satisfacible para $I(p) = 1, I(q) = 0$

2.3.5 Selección de tautologías

1	$F \to F$	Ley de identidad
2	$F \lor \neg F$	Ley del tercio excluido
3	$\neg(F \land \neg F)$	Principio de no contradicción
4	$(\neg F \to F) \to F$	Ley de Clavius
5	$\neg F \to (F \to G)$	Ley de Duns Scoto
6	$((F \to G) \to F) \to F$	Ley de Peirce
7	$(F \to G) \land F \to G$	Modus ponens
8	$(F \to G) \land \neg G \to \neg F$	Modus tollens

2.3.6 Fórmulas equivalentes

Definición

F y G son **equivalentes** si $I(F) = I(G)$ para toda interpretación I

$$F \equiv G$$

Ejemplo 7

Ejemplos de equivalencias:

- Idempotencia: $F \lor F \equiv F$; $F \land F \equiv F$
- Conmutatividad: $F \lor G \equiv G \lor F$; $F \land G \equiv G \land F$.
- Asociatividad: $F \lor (G \lor H) \equiv (F \lor G) \lor H$; $F \land (G \land H) \equiv (F \land G) \land H$
- Absorción: $F \land (F \lor G) \equiv F$; $F \lor (F \land G) \equiv F$
- Distributividad: $F \land (G \lor H) \equiv (F \land G) \lor (F \land H)$; $F \lor (G \land H) \equiv (F \lor G) \land (F \lor H)$
- Doble negación: $\neg\neg F \equiv F$
- Leyes de De Morgan: $\neg(F \land G) \equiv \neg F \lor \neg G$; $\neg(F \lor G) \equiv \neg F \land \neg G$
- Leyes de tautologías: Si F es una tautología, $F \land G \equiv G$; $F \lor G \equiv F$
- Leyes de contradicciones: Si F es una contradicción $F \land G \equiv F$; $F \lor G \equiv G$

☞ ***Propiedades de la equivalencia lógica***

- Relación entre equivalencia y bicondiconal:

  $$F \equiv G \text{ syss (si y solo si) } \models F \leftrightarrow G$$

- Propiedades básicas de la equivalencia lógica:
  - Reflexiva: $F \equiv F$
  - Simétrica: Si $F \equiv G$, entonces $G \equiv F$
  - Transitiva: Si $F \equiv G$ y $G \equiv H$, entonces $F \equiv H$
- Principio de sustitución de fórmulas equivalentes:
  Si en la fórmula F se sustituye una de sus subfórmulas G por una fórmula G' lógicamente equivalente a G, entonces la fórmula obtenida, F', es lógicamente equivalente a F.
  Ejemplos:
  - $F = \neg(p \land q) \to \neg(p \land \neg\neg r)$
  - $G = \neg(p \land q)$
  - $G' = \neg p \lor \neg q$
  - $F' = (\neg p \neg q) \to \neg(p \land \neg\neg r)$

2.3.7 Modelo de conjuntos de fórmulas

Definición

$S, S_1, S_2, \ldots$ representarán conjuntos de fórmulas. I **es modelo de** S si para toda $F \in S$ se tiene que $I \models F$.

$$I \models S$$

Ejemplo 8

Sea $S = \{(p \vee q) \wedge (\neg q \vee r), q \rightarrow r\}$

– La interpretación I_1 tal que $I_1(p) = 1$, $I_1(q) = 0$, $I_1(r) = 1$ es modelo de $S (I_1 \models S)$.

$$\begin{array}{cccc} \{(p \vee q) & \wedge & (\neg q \vee r), & q \rightarrow r\} \\ \{(1 \vee 0) & \wedge & (\neg 0 \vee 1), & 0 \rightarrow 1\} \\ \{1 & \wedge & (1 \vee 1), & 1\} \\ \{1 & \wedge & 1, & 1\} \\ \{ & 1, & & 1\} \end{array}$$

La interpretación I_2 tal que $I_2(p) = 0$, $I_2(q) = 1$, $I_1(r) = 0$ no es modelo de $S (I_2 \not\models S)$.

$$\begin{array}{cccc} \{(p \vee q) & \wedge & (\neg q \vee r), & q \rightarrow r\} \\ \{(0 \vee 1) & \wedge & (\neg 1 \vee 0), & 1 \rightarrow 0\} \\ \{1 & \wedge & (0 \vee 0), & 0\} \\ \{1 & \wedge & 0, & 0\} \\ \{ & 0, & & 0\} \end{array}$$

2.3.8 Conjunto consistente de fórmulas

Definición

S **es consistente** si S tiene algún modelo. Por otro lado, S es **inconsistente** si S no tiene ningún modelo.

Ejemplo 9

De acuerdo con la siguiente tabla de verdad:

	p	q	r	$(p \vee q)$	$(\neg q \vee r)$	$(p \vee q) \wedge (\neg q \vee r)$	$p \rightarrow r$	$\neg r$
I_1	0	0	0	0	1	0	1	1
I_2	0	0	1	0	1	0	1	0
I_3	0	1	0	1	0	0	1	1
I_4	0	1	1	1	1	1	1	0
I_5	1	0	0	1	1	1	0	1
I_6	1	0	1	1	1	1	1	0
I_7	1	1	0	1	0	0	0	1
I_8	1	1	1	1	1	1	1	0

- $\{(p \vee q) \wedge (\neg q \vee r), p \rightarrow r\}$ es consistente con los modelos I_4, I_6, I_8.
- $\{(p \vee q) \wedge (\neg q \vee r), p \rightarrow r, \neg r\}$ es inconsistente.

2.3.9 Consecuencia lógica

Definición

F **es consecuencia de** S si todos los modelos de S son modelos de F

$$S \models F$$

Ejemplo 10

De acuerdo con la siguientes tablas de verdad, $\{p \to q, q \to r\} \models p \to r$ y $\{p\} \not\models p \land q$

	p	q	r	$p \to q$	$q \to r$	$p \to r$
I_1	0	0	0	1	1	1
I_2	0	0	1	1	1	1
I_3	0	1	0	1	0	1
I_4	0	1	1	1	1	1
I_5	1	0	0	0	1	0
I_6	1	0	1	0	1	1
I_7	1	1	0	1	0	0
I_8	1	1	1	1	1	1

p	q	$p \land q$
1	1	1
1	0	0
0	1	0
0	0	0

☞ ***Propiedades de la consecuencia***

- Propiedades básicas de la relación de consecuencia:
  * Reflexividad: $S \models S$
  * Monotonía: Si $S_1 \models F$ y $S_1 \subseteq S_2$, entonces $S_2 \models F$
  * Transitividad: Si $S \models F$ y $\{F\} \models G$, entonces $S \models G$
- Relación entre consecuencia, validez, satisfacibilidad y consistencia. Las siguientes condiciones son equivalentes:
  1. $\{F_1, \dots, F_n\} \models G$
  2. $\models F_1 \land \cdots \land F_n \to G$
  3. $\neg(F_1 \land \cdots \land F_n \to)$ es insatisfacible
  4. $\{F - 1, \dots, F_n, \neg G\}$ es inconsistente

2.3.10 Argumentaciones y problemas lógicos

El siguiente es un ejemplo de argumentación:

Problema de los animales. Se sabe que

- Los animales con pelo o que dan leche son mamíferos.
- Los mamíferos que tienen pezuñas o que rumian son ungulados.
- Los ungulados de cuello largo son jirafas.
- Los ungulados con rayas negras son cebras.

Se observa un animal que tiene pelos, pezuñas y rayas negras. Por consiguiente, se concluye que el animal es una cebra.

El argumento se puede formalizar de la siguiente manera:

Formalización

{ tiene_pelos ∨ da_leche → es_mamífero,

es_mamífero ∧ (tiene_pezuñas ∨ rumia) → es_ungulado,

es_ungulado ∧ tiene_cuello_largo → es_jirafa,

es_ungulado ∧ tiene_rayas_negras → es_cebra,

tiene_pelos ∧ tiene_pezuñas ∧ tiene_rayas_negras }

$\models$ *es_cebra*

A continuación se muestra un ejemplo de problemas lógicos: **Veraces y mentirosos**. En una isla hay dos tribus, la de los veraces (que siempre dicen la verdad) y la de los mentirosos (que siempre mienten). Un viajero se encuentra con tres isleños A, B y C y cada uno le dice una frase

- A dice "B y C son veraces syss C es veraz"
- B dice "Si A y C son veraces, entonces B y C son veraces y A es mentiroso"
- C dice "B es mentiroso syss A o B es veraz"

Determinar a qué tribu pertenecen A, B y C.
Simbolización.

- a: "A es veraz", b: "B es veraz", c: "C es veraz".

Formalización.

- $F_1 = a \leftrightarrow (b \wedge c \leftrightarrow c)$, $F_2 = b \leftrightarrow (a \wedge c \rightarrow b \wedge c \wedge \neg a)$ y $F_3 = c \leftrightarrow (\neg b \leftrightarrow a \vee b)$.

Modelos de $F_1, F_2 y F_3$

- Si I es modelo de $\{F_1, F_2, F_3\}$, entonces $I(a) = 1, I(b) = 1, I(c) = 0$.

Conclusión: A y B son veraces y C es mentiroso.

3 Deducción en lógica proposicional

3.1 Reglas de deducción

3.1.1 Reglas de la conjunción

- Regla de introducción de la conjunción:

$$\frac{F \qquad G}{F \wedge G} \wedge_i$$

- Reglas de eliminación de la conjunción:

$$\frac{F \wedge G}{F} \wedge_{e_1} \qquad \frac{F \wedge G}{G} \wedge_{e_2}$$

Ejemplo 1

$p \wedge q,\, r \vdash q \wedge r$:

1.	$p \wedge q$	premisa
2.	r	premisa
3.	q	$\wedge_{e_2}$ 1
4.	$q \wedge r$	$\wedge_i$ 2, 3

Adecuación de las reglas de la conjunción:

$$\wedge_i : \{F, G\} \models F \wedge G$$
$$\wedge_{e_1} : F \wedge G \models F$$
$$\wedge_{e_2} : F \wedge G \models G$$

3.1.2 Reglas de la doble negación

- Regla de eliminación de la doble negación:

$$\frac{\neg\neg F}{F}\ \neg\neg e$$

- Regla de introducción de la doble negación:

$$\frac{F}{\neg\neg F}\ \neg\neg i$$

Ejemplo 2

$p, \neg\neg(q \wedge r) \vdash \neg\neg p \wedge r$:

1.	p	premisa
2.	$\neg\neg(q \wedge r)$	premisa
3.	$\neg\neg p$	$\neg\neg i$ 1
4.	$q \wedge r$	$\neg\neg e$ 2
5.	r	$\wedge_{e_3} 4$
6.	$\neg\neg p \wedge r$	$\wedge_i$ 3, 5

Adecuación de las reglas de la conjunción:

$$\neg\neg e : \{\neg\neg F\} \models F$$
$$\neg\neg i : \{F\} \models \neg\neg F$$

3.1.3 Reglas de eliminación del condicional

- Regla de eliminación del condicional:

$$\frac{F \quad F \to G}{G}\ \to e$$

Ejemplo 3

$\neg p \wedge q, \neg p \wedge q \to r \vee \neg p \vdash r \vee \neg p$:

1.	$\neg p \wedge q$	premisa
2.	$\neg p \wedge q \to r \vee \neg p$	premisa
3.	$r \vee \neg p$	$\to e$ 1, 2

Ejemplo 4

$p, p \to q, p \to (q \to r)$:

1.	p	premisa
2.	$p \to q$	premisa
3.	$p \to (q \to r)$	premisa
4.	q	$\to e$ 1, 2
5.	$q \to r$	$\to e$ 1, 3
6.	r	$\to e$ 4, 5

3.1.4 Regla derivada de modus tollens

- Regla derivada de modus tollens:

$$\frac{F \to F \quad \neg G}{\neg F}\ MT$$

Ejemplo 5

$p \to (q \to r), p, \neg r \vdash \neg q$:

1.	$p \to (q \to r)$	premisa
2.	p	premisa
3.	$\neg r$	premisa
4.	$q \to r$	$\to e$ 1, 2
5.	$\neg q$	MT 3, 4

Ejemplo 6

$\neg p \to q, \neg q \vdash p$:

1.	$\neg p \to q$	premisa
2.	$\neg q$	premisa
3.	$\neg\neg p$	MT 1, 2

3.1.5 Regla de introducción del condicional

- Regla de introducción del condicional:

$$\frac{\begin{array}{c} \hline F \\ \cdot \\ \cdot \\ \cdot \\ G \\ \hline \end{array}}{F \to G}\ \to i$$

Ejemplo 7

$p \to q \vdash \neg q \to \neg p$:

1.	$p \to q$	premisa
2.	$\neg q$	supuesto
3.	$\neg p$	MT 1, 2
4.	$\neg q \to \neg p$	$\to i$ 2 - 3

Adecuación de la regla de introducción del condicional

Si $F \models G$, entonces $\models F \to G$

Ejemplo 8

$\neg q \to \neg p \vdash p \to \neg\neg q$:

1.	$\neg q \to \neg p$	premisa
2.	p	supuesto
3.	$\neg\neg p$	$\neg\neg i$ 2
4.	$\neg\neg q$	MT 1, 3
5.	$p \to \neg\neg q$	$\to i$ 2 - 4

Ejemplo 9

$(q \to r) \to ((\neg q \to \neg p) \to (p \to r))$:

1.	$q \to r$	supuesto
2.	$\neg p \to \neg p$	supuesto
3.	p	$i$supuesto
4.	$\neg\neg p$	$\neg\neg i$ 3
5.	$\neg\neg q$	MT 2, 3
6.	q	$\neg\neg e$ 5
7.	r	$\to e$ 1, 6
8.	$p \to r$	$\to i$ 3 - 7
9.	$(\neg q \to \neg p) \to (p \to r)$	$\to i$ 2 - 8
10.	$(q \to r) \to ((\neg q \to \neg p) \to (p \to r))$	$\to i$ 1 - 9

3.1.6 Reglas de la disyunción

- Reglas de introducción de la disyunción

$$\frac{F}{F \vee G} \;\; \vee i_1$$

$$\frac{G}{F \vee G} \;\; \vee i_2$$

- Regla de eliminación de la disyunción:

$$\begin{array}{c} \begin{array}{ccc} & \overline{F} & \overline{G} \\ & \vdots & \vdots \\ F \vee G & \underline{H} & \underline{H} \end{array} \\ \hline H \end{array} \;\; \vee e$$

Ejemplo 10

$q \to r \vdash p \vee q \to p \vee r$:

1.	$q \to r$	premisa
2.	$p \vee q$	supuesto
3.	p	supuesto
4.	$p \vee r$	$\vee r_1$ 3
5.	q	**supuesto**
6.	r	$\to e$ 1, 5
7.	$p \vee r$	$\vee i_2$ 6
8.	$p \vee r$	$\vee e$ 2, 3 - 4, 5 - 7
9.	$p \vee q \to p \vee r$	$\to i$ 2 - 8

3.1.7 Reglas de la negación

Extensiones de la lógica para usar falso:

- Extensión de la sintasis: $\perp$ es una fórmula proposicional.
- Extensión de la semántica: $I(\perp) = 0$ en cualquier interpretación I.

Las siguientes son reglas de la negación:

- Regla de eliminación de lo falso:

$$\frac{\perp}{F} \perp e$$

- Regla de eliminación de la negación:

$$\frac{F \qquad \neg F}{\perp} \neg e$$

Adecuación de las reglas de la negación:

- $\perp \models F$
- $\{F, \neg F\} \models \perp$

Ejemplo 11

$\neg p \vee q \vdash p \rightarrow q$:

1.	$\neg p \vee q$	premisa
2.	p	supuesto
3.	$\neg p$	supuesto
4.	$\perp$	$\neg e$ 2, 3
5.	q	$\perp e$ 4
6.	q	supuesto
7.	q	$\vee e$ 1, 3 - 5, 6 - 6
8.	$p \rightarrow q$	$\rightarrow i$ 2 - 7

- Regla de introducción de la negación:

$$\frac{\begin{array}{c} F \\ \cdot \\ \cdot \\ \cdot \\ \perp \end{array}}{\neg F} \neg i$$

Adecuación : Si $F \models \perp$, entonces $\models \neg F$

Ejemplo 12

$p \to q, p \to \neg q \vdash \neg p$:

1.	$p \to q$	premisa
2.	$p \to \neg q$	premisa
3.	p	supuesto
4.	q	$\to e$ 1, 3
5.	$\neg q$	$\to e$ 2, 3
6.	$\perp$	$\neg e$ 4, 5
7.	$\neg p$	$\neg i$ 3 - 6

3.1.8 Regla del bicondicional

- Regla de introducción del bicondicional:

$$\frac{F \to G \quad G \to F}{F \leftrightarrow G} \leftrightarrow i$$

Ejemplo 13

$p \wedge q \leftrightarrow q \wedge p$:

1.	$p \wedge q$	supuesto
2.	p	$\wedge e_1$ 1
3.	q	$\wedge e_2$ 1
4.	$q \wedge p$	$\wedge i$ 2,3
5.	$p \wedge q \to q \wedge p$	$\to i$ 1 - 4
6.	$q \wedge p$	supuesto
7.	q	$\wedge e_2$ 6
8.	p	$\wedge e_1$ 6
9.	$p \wedge q$	$\wedge i$ 7, 8
10.	$q \wedge p \to p \wedge q$	$\to i$ 6 - 9

- Eliminación del bicondicional:

$$\frac{F \leftrightarrow G}{F \to G} \leftrightarrow e_1$$

$$\frac{F \leftrightarrow G}{G \to F} \leftrightarrow e_2$$

Ejemplo 14

$p \wedge q \leftrightarrow q \wedge p$:

1.	$p \leftrightarrow q$			premisa
2.	$p \vee q$			premisa
3.	p	supuesto	q	supuesto
4.	$p \to q$	$\leftrightarrow e_1$ 1	$q \to p$	$\to e_2 1$
5.	$q \to q$	$\to e$ 4, 3	p	$\to e$ 4', 3'
6.	$p \wedge q$	$\wedge i$ 3, 5	$p \wedge q$	$\wedge i$ 3', 5'
7.	$p \wedge q$			$\wedge e$ 2, 3 - 6-, 3' - 5'

3.2 Reglas derivadas

3.2.1 Regla del modus tollens

- Regla derivada de modus tollens (MT):

$$\frac{F \leftrightarrow G \quad \neg G}{\neg F}\; MT$$

Derivación:

1.	$F \rightarrow G$	premisa
2.	$\neg G$	premisa
3.	F	supuesto
4.	G	$\rightarrow e$ 1,3
5.	$\bot$	$\neg e$ 2 - 4
6.	$\neg F$	$\neg i$ 2 - 4

3.2.2 Regla de introducción de la doble negación

- Regla de introducción de la doble negación:

$$\frac{F}{\neg\neg F}\; \neg\neg i$$

Derivación:

1.	F	premisa
2.	$\neg F$	supuesto
3.	$\bot$	$\neg e$ 1, 2
4.	$\neg\neg F$	$\neg i$ 2 - 3

3.2.3 Regla de reducción al absurdo (RAA)

- Regla de reducción al absurdo:

$$\frac{\boxed{\begin{array}{c} \neg F \\ \cdot \\ \cdot \\ \cdot \\ \bot \end{array}}}{F}\; RAA$$

Derivación:

1.	$\neg F \rightarrow \bot$	premisa
2.	$\neg F$	supuesto
3.	$\bot$	$\rightarrow e$ 1, 2
4.	$\neg\neg F$	$\neg i$ 2 - 3
5.	F	$\neg e$ $\neg$4

3.2.4 Ley del tercio exluido (LEM)

- Ley del tercio exluido (LEM):

$$\frac{}{F \lor \neg F}\ LEM$$

Derivación:

1.	$\neg(F \lor \neg F)$	supuesto
2.	F	supuesto
3.	$F \lor \neg F$	$\lor i_1$ 2
4.	$\bot$	$\neg e$ 1, 3
5.	$\neg F$	$\neg i$ 2 - 4
6.	$F \lor \neg F$	$\lor i_2$ 5
7.	$\bot$	$\neg e$ 1, 6
8.	$F \lor \neg F$	RAA 1 - 7

Ejemplo 15

$p \to q \vdash \neg p \lor q$:

1.	$p \to q$	premisa
2.	$p \lor \neg p$	LEM
3.	p	supuesto
4.	q	$\to e$ 1, 3
5.	$\neg p \lor q$	$\lor i_2$ 4
6.	$\neg p$	supuesto
7.	$\neg p \lor q$	$\lor i_1$ 6
8.	$\neg p \lor q$	$\lor e$ 2, 3 - 5, 6 - 7

4 Sintáxis y semántica de primer orden

4.1 Representación del conocimiento en lógica de primer orden

4.1.1 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:

$$\{SvC \to CvS, SvC\} \models CvS$$

Ejemplo 1

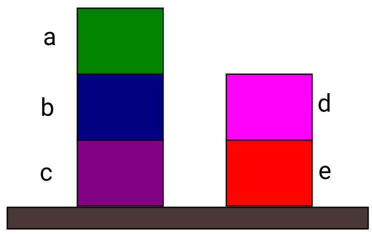
Si una ciudad es vecina de otra, entonces la segunda es vecina de la primera. Sevilla es vecina de Cádiz. Por tanto, Cádiz es vecina de Sevilla

Representación en lógica proposicional: Imposible

Representación en lógica de primer orden:

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

4.1.2 Representación del 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: $sobre(a, b)$, $sobre(b, c)$, $sobre\_mesa(c)$, $sobre(d, e)$, $sobre\_mesa(e)$
Definiciones:

- $bajo(x, y)$ se verifica si el bloque x está debajo del bloque y $\forall x \forall y[bajo(x, y) \leftrightarrow sobre(y, x)]$
- $encima(x, y)$ se verifica si el bloque x está encima del bloque y pudiendo haber otros bloques entre ellos
$$\forall x \forall y[encima(x, y) \leftrightarrow sobre(x, y) \vee \exists z[sobre(x, z) \wedge encima(z, y)]]$$
- $libre(x)$ se verifica si el bloque x no tiene bloques encima
$$\forall x[libre(x) \rightarrow \neg \exists y sobre(y, x)]$$
- $pila(x, y, z)$ se verifica si el bloque x está sobre el y , el y sobre el z y el z sobre la mesa
$$\forall x \forall y \forall z[pila(x, y, z) \leftrightarrow sobre(x, y) \wedge sobre(y, z) \wedge sobre\_mesa(z)]$$

☞ ***Propiedades:*** Si z, y, z es una pila entonces y no está libre

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

4.1.3 Representación del mundo de los bloques con funciones e igualdad

- Simbolización:
  - $es\_bloque(x)$ se verifica si x es un bloque.
  - $superior(x)$ es el bloque que está sobre el bloque x.
- Situación del ejemplo:
  - $es\_bloque(a)$, $es\_bloque(b)$, $es\_bloque(c)$, $es\_bloque(d)$, $es\_bloque(e)$
  - $superior(b) = a, superior(c) = b, superior(e) = d$
- Definiciones:

– $sobre_mesa(x)$ se verifica si el bloque x está sobre la mesa

$$\forall x[sobre\_mesa(x) \leftrightarrow es\_bloque(x) \wedge \neg\exists y superior(y) = x]$$

– $libre(x)$ se verifica si el bloque x no tiene bloques encima

$$\forall x[libre(x) \rightarrow \neg\exists y superior(x) = y]$$

– $tope(x)$ es el bloque libre que está encima de x

$$\forall x[(libre(x) \rightarrow tope(x) = x) \wedge (\neg libre(x) \rightarrow tope(x) = tope(superior(x)))]$$

4.1.4 Representación de conocimiento astronómico

- La Tierra es un planeta:
$$planeta(Tierra)$$
- La Luna no es un planeta:
$$\neg planeta(Luna)$$
- La Luna es un satélite:
$$sat\acute{e}lite(Luna)$$
- La Tierra gira alrededor del Sol:
$$gira(Tierra, Sol)$$
- Todo planeta es un satélite:
$$\forall x[planeta(x) \rightarrow \ sat\acute{e}lite(x)]$$
- Todo planeta gira alrededor del Sol:
$$\forall x[planeta(x) \rightarrow gira(x, Sol)]$$
- Algún planeta gira alrededor de la Luna:
$$\exists x[planeta(x) \wedge gira(x, Luna)]$$
- Hay por lo menos un satélite:
$$\exists x \ sat\acute{e}lite(x)$$
- Ningún planeta es un satélite:
$$\neg\exists x[planeta(x) \wedge \ sat\acute{e}lite(x)]$$
- Ningún objeto celeste gira alrededor de sí mismo:
$$\neg\exists x gira(x, x)$$
- Alrededor de los satélites no giran objetos:
$$\forall x[\ sat\acute{e}lite(x) \rightarrow \neg\exists y gira(y, x)]$$
- Hay exactamente un satélite:
$$\exists x[\ sat\acute{e}lite(x) \wedge \forall y[\ sat\acute{e}lite(y) \rightarrow x = y]]$$
- La Luna es un satélite de la Tierra:
$$sat\acute{e}lite(Luna, Tierra)$$
- Todo planeta tiene un satélite:
$$\forall x[planeta(x) \rightarrow \exists y \ sat\acute{e}lite(y, x)]$$

4.2 Sintaxis de la lógica de primer orden

4.2.1 Lenguaje de primer orden

Símbolos lógicos:

- Variables: $x, y, z, \dots, x1, x2, \dots$.
- 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.

Ejemplo 2

Lenguaje del mundo de los bloques:

- Símbolos de constantes: a, b, c, d, e
- Símbolos de predicado (y de relación):
  - de aridad 1: *sobre_mesa*, *libre*, *es_bloque*
  - de aridad 2: *sobre*, *bajo*, *encima*
  - de aridad 3: *pila*
- Símbolos de función (de aridad 1): *superior*, *tope*

Ejemplo 3

Lenguaje de la aritmética:

- Símbolos de constantes: $0, 1$
- Símbolos de función:
  - monaria: s (siguiente)
  - binarias: $+, \cdot$
- Símbolo de predicado binario: $<$

4.2.2 Términos

Definición

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.

Ejemplo 4

- 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,
  - $superior(superior(c))$ es un término.
  - $libre(superior(c))$ no es un término.

Notación:

- $s, t, t_1, t_2, \dots$ representan términos.
- $Térm(L)$ representa el conjunto de los términos de L.

4.2.3 Fórmulas atómicas

Definición

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.

Ejemplo 5

- 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,
  - $libre(superior(c))$ es una fórmula atómica.
  - $tope(c) = superior(b)$ es una fórmula atómica.

Notación:

- $A, B, A_1, A_2, \ldots$ representan fórmulas atómicas.
- $Atóm(L)$ representa el conjunto de las fórmulas atómicas de L.

Definición

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.

Ejemplo 6

- 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,
  - $\forall x(tope(x) = x \leftrightarrow libre(x))$ es una fórmula.

Notación:

- $F, G, H, F_1, F_2, \ldots$ representan fórmulas.
- $Fórm(L)$ representa el conjunto de las fórmulas de L.

4.2.4 Subfórmulas

Definición

El conjunto $Subf(F)$ de las subfórmulas de una fórmula F se define recursivamente por:

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

Ejemplo 7

$$\begin{aligned} Subf(\forall x(R(x,c) \rightarrow P(f(y)))) = \quad & \{\forall x(R(x,c) \rightarrow P(f(y))), \\ & (R(x,c) \rightarrow P(f(y))), \\ & R(x,c), \\ & P(f(y))\} \end{aligned}$$

4.2.5 Criterios de reducción de paréntesis

- Pueden eliminarse los paréntesis externos.

$$F \wedge G \text{ es una abreviatura de } (F \wedge G)$$

- Precedencia de asociación de conectivas y cuantificadores:

$$\forall, \exists, \neg, \wedge, \vee, \to, \leftrightarrow$$
$$\forall x P(x) \to Q(x) \text{ es una abreviatura de } (\forall x P(x)) \to Q(x)$$

- Cuando una conectiva se usa repetidamente, se asocia por la derecha.

$$F \vee G \vee H \text{ es una abreviatura de } (F \vee (G \vee H))$$
$$F \wedge G \wedge H \to \neg F \vee G \text{ es una abreviatura de } ((F \wedge (G \wedge H)) \to (\neg F \vee G)$$

- Los símbolos binarios pueden escribirse en notación infija.

$$x + y \text{ es una abreviatura de } +(x, y)$$
$$x < y \text{ es una abreviatura de } <(x, y)$$

4.2.6 Conjuntos de variables

Definición

El conjunto de las variables del término t es

$$V(t) = \begin{cases} \emptyset, & \text{si } t \text{ es una constante;} \\ \{x\} & \text{si } t \text{ es una variable} x; \\ V(t_1) \cup \cdots \cup V(t_n), & \text{si } t \text{ es} f(t_1, \dots, t_n) \end{cases}$$

Definición

El conjunto de las variables de la fórmula F es

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

Ejemplo 8

- El conjunto de las variables de $\forall x(R(x, c) \to P(f(y)))$ es $\{x, y\}$.
- El conjunto de las variables de $\forall x(R(a, c) \to P(f(y)))$ es $\{y\}$.

4.2.7 Variables libres y ligadas

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

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

Ejemplo 9

Table 2: My caption

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

4.2.8 Fórmulas cerradas y abiertas

Definición

Una **fórmula cerrada** (o sentencia) es una fórmula sin variables libres.

Ejemplo 10

- $\forall x(P(x) \to \exists y R(x, y))$ es cerrada.
- $\exists x R(x, y) \vee \forall y P(y)$ no es cerrada.

Definición

Una **fórmula abierta** es una fórmula con variables libres.

Ejemplo 11

- $\forall x(P(x) \to \exists y R(x, y))$ no es abierta.
- $\forall x R(x, y) \vee \forall y P(y)$ es abierta.

4.3 Semántica de la lógica de primer orden

4.3.1 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 $I(f) : U^n \to U$;
  - 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 $A : Var \to U$ 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.

Ejemplo 12

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)
  - $I_2(0) =$ (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)
  - $I_2(\leq) = \{(w_1, w_2) : w_1, w_2 \in \{0, 1\}^*, w_1$ es prefijo de $w_2\}$ (prefijo)
- Tercera estructura de L:
  - $U_3 = \{abierto, cerrado\}$
  - $I_3(0) = cerrado$
  - $I_3(s) = \{(abierto, cerrado), (cerrado, abierto)\}$
  - $I_3(+) = \{(abierto, abierto, abierto), (abierto, cerrado, abierto), (cerrado, abierto, abierto), (cerrado, cerrado$
  - $I_3(\leq) = \{(abierto, abierto), (cerrado, abierto), (cerrado, cerrado)\}$

e	$I_3(s)(e)$
abierto	cerrado
cerrado	abierto

$I_3(+)$	abierto	cerrado
abierto	abierto	abierto
cerrado	abierto	cerrado

$I_3(\leq)$	abierto	cerrado
abierto	1	0
cerrado	1	1

4.3.2 Evaluación de términos

Definición

Dada una estructura $\mathcal{I} = (U, I)$ de L y una asignación A en $\mathcal{I}$, se define la **función de evaluación de términos** $\mathcal{I}_A : \mathit{Térm}(L) \to U$ 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 13

Sean L el lenguaje del Ejemplo 12, t el término $s(+(x, s(0)))$, $\mathcal{I}$ la primera estructura y $A(x) = 3$.

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

4.3.3 Evaluación de fórmulas

Definición

Dada una estructura $\mathcal{I} = (U, I)$ de L y una asignación A sobre $\mathcal{I}$, se define la **función de evaluación de fórmulas** $\mathcal{I}_A : \mathit{Fórm}(L) \to B$ por

$$\begin{array}{cc} \text{Si } F \text{ es } t_1 = t_2, & \mathcal{I}_A(F) = H = (\mathcal{I}_A(t_1), \mathcal{I}_A(t_2)) \\ \text{Si } F \text{ es } P(t_1, \dots, t_n), & \mathcal{I}_A(F) = H_{I(P)}(\mathcal{I}_A(t_1), \dots, \mathcal{I}_A(t_n)) \\ \text{Si } F \text{ es } \neg G, & \mathcal{I}_A(F) = H\neg(\mathcal{I}_A(G)) \\ \text{Si } F \text{ es } G * H, & \mathcal{I}_A(F) = H * (\mathcal{I}_A(G), \mathcal{I}_A(H)) \end{array}$$

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 $\exists x G$,

$$\mathcal{I}_A(F) = \begin{cases} 1, & \text{Si para todo } u \in U \text{ tal que } \mathcal{I}_{A[x/u]}(G) = 1; \\ 0, & \text{en caso contrario} \end{cases}$$

4.3.4 Conceptos auxilares para la evaluación de fórmulas

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

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

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

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

Ejemplo 14

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

$$\begin{array}{ccc}
\mathcal{I}_A(\forall x \exists y P(x, y)) & = & V \Leftrightarrow \mathcal{I}_{A[x/1]}(\exists y P(x, y)) = V y \\
& & \mathcal{I}_{A[x/2]}(\exists y P(x, y)) = V \\
\mathcal{I}_{A[x/1]}(\exists y P(x, y)) & = & V \Leftrightarrow \mathcal{I}_{A[x/1,y/1]} P(x, y) = V \text{ó} \\
& & \mathcal{I}_{A[x/1,y/2]} P(x, y) = V \\
\mathcal{I}_{A[x/1,y/1]} P(x, y) = P^I(1, 1) = V & & \\
\text{Luego, } \mathcal{I}_{A[x/1]}(\exists y P(x, y)) = V. & & \\
\mathcal{I}_{A[x/2]}(\exists y P(x, y)) & = & V \Leftrightarrow \mathcal{I}_{A[x/2,y/1]} P(x, y) = V \text{ ó} \\
& & \mathcal{I}_{A[x/2,y/2]} P(x, y) = V \\
\mathcal{I}_{A[x/2,y/2]} P(x, y) = P^I(2, 2) = V & & \\
\text{Luego,} \mathcal{I}_{A[x/2]}(\exists y P(x, y)) = V. & & \\
\text{Por tanto,} \mathcal{I}_A(\forall x \exists y P(x, y)) = V & &
\end{array}$$

4.3.5 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 $\mathcal{I}(F)$.

4.3.6 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 es 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 todo asignación A en $\mathcal{I}, \mathcal{I}_A(F) = 1$. Se representa por $\mathcal{I} \models F$.

Ejemplo 15

Sea $\mathcal{I} = (\mathbb{N}, I)$ una estructura tal que $I(f) = +eI(g) = *$.

- 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 $\mathcal{I}_B \models f(x, y) = g(x, y)$,
- $\mathcal{I} \models f(x, y) = g(x, y)$
- $\mathcal{I} \models f(x, y) = f(y, x)$

4.3.7 Satisfacibilidad y validez

Definición

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 $\mathcal{I}_A(F) = 0$).

Ejemplo 16

- $\exists x P(x) \vee \forall x \neg P(x)$ es válida.
- $\exists x P(x) \wedge \exists x \neg P(x)$ es satisfacible, pero no es válida.
- $\forall x P(x) \wedge \exists x \neg P(x)$ es insatisfacible.
- F es válida syss $\neg F$ es insatisfacible.

  F es válida
  $\Leftrightarrow$ para toda estructura $\mathcal{I}$ y toda asignación A se tiene que $\mathcal{I}_A(F) = 1$
  $\Leftrightarrow$ para toda estructura $\mathcal{I}$ y toda asignación A se tiene que $\mathcal{I}_A(\neg F) = 0$
  $\Leftrightarrow$ $\neg F$ es insatisfacible.

- Si F es válida, entonces F es satisfacible.

  F es válida
  $\Rightarrow$ para toda estructura $\mathcal{I}$ y toda asignación A se tiene que $\mathcal{I}_A(F) = 1$
  $\Rightarrow$ existe una estructura $\mathcal{I}$ y una asignación A tales que $\mathcal{I}_A(F) = 1$
  $\Rightarrow$ F es satisfacible.

- F es satisfacible $\Rightarrow \neg F$ es insatisfacible.

  $\forall x P(x)$ y $\neg\forall x P(x)$ son satisfacibles.

- Sea F una fórmula de L y $x_1, \ldots, x_n$ las variables libres de F.

  F es válida syss $\forall x_1 \ldots \forall x_n F$ es válida.
  [$\forall x_1 \ldots \forall x_n F$ es el cierre universal de F].
  F es satisfacible syss $\exists x_1 \ldots \exists x_n F$ es satisfacible.
  [$\exists x_1 \ldots \exists x_n F$ es el cierre existencial de F].

4.3.8 Modelo de un conjunto de fórmulas

Notación: $S, S1, S2, \ldots$ representarán conjuntos de fórmulas.

Definición

Sean S un conjunto de fórmulas de L, $\mathcal{I}$ 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 $\mathcal{I} \models S$.

Ejemplo 17

- 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.

Ejemplo 18

- 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.

4.3.9 Consistencia de un conjunto de fórmulas

Definición

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 $F \in S, \mathcal{I}_A(F) = 1$).
- 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 $\mathcal{I}_A(F) = 0$).

Ejemplo 19

- $S = \{\forall y R(x,y), \forall y f(x,y) = y\}$ es consistente. $(\mathcal{I}, A)$ con $\mathcal{I} = (\mathbb{N}, I), R^I = \leq, f^I = +, A(x) = 0$ es realización de S.
- $S = \{P(x) \to Q(x), \forall y P(y), \neg Q(x)\}$ es inconsistente.

☞ ***Propiedad.*** Sea S un conjunto de fórmulas cerradas de L. Entonces S es consistente syss S tiene algún modelo.

4.3.10 Consecuencia lógica

Definición

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 $\{G\} \not\models F$.

Ejemplo 20

- $\forall x P(x) \models P(y)$
- $P(y) \not\models \forall x P(x)$
  $(\mathcal{I}, A)$ con $\mathcal{I} = (U, I), U = 1, 2, P^I = 1, A(y) = 1$.
- $\{\forall x (P(x) \to Q(x)), P(c)\} \models= Q(c)$
- $\{\forall x (P(x) \to Q(x)), Q(c)\} \not\models P(c)$
  $(\mathcal{I}, A)$ con $\mathcal{I} = (U, I), U = 1, 2, c^I = 1, P^I = 2, Q^I = 1, 2$.
- $\{\forall x (P(x) \to Q(x)), \neg Q(c)\} \models \neg P(c)$
- $\{P(c), \neg P(d)\} \models c \neq d$

4.3.11 Consecuencia lógica e inconsistencia

$S \models F syss S \cup \{\neg F\}$ es inconsistente.

$S \models F$

$\Leftrightarrow$ 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 $\mathcal{I}_A(F) = 1$.

$\Leftrightarrow$ 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 $\mathcal{I}_A(\neg F) = 0$.

$\Leftrightarrow$ para toda estructura I de L y toda asignación A en $\mathcal{I}$, existe alguna $H \in S \cup \{\neg F\}$ tal que $\mathcal{I}_A(H) = 0$.

$\Leftrightarrow S \cup \{\neg F\}$ 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.

4.3.12 Equivalencia lógica

Definición

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 $F \equiv G$.

Ejemplo 20

- $P(x) \equiv P(y)$.
  $\mathcal{I} = (\{1, 2\}, I)$ con $P^I = \{1\} y A(x) = 1, A(y) = 2$.
- $\forall x P(x) \equiv \forall y P(y)$.
- $\forall x(P(x) \land Q(x)) \equiv \forall x P(x) \land \forall x Q(x)$.
- $\exists x(P(x) \land Q(x)) \equiv \exists x P(x) \land \exists x Q(x)$.
  $\mathcal{I} = (\{1, 2\}, I)$ con $P^I = \{1\}$ y $Q^I = \{2\}$.

☞ ***Propiedades básicas de la equivalencia lógica***:
  - Reflexiva: $F \equiv F$
  - Simétrica: Si $F \equiv G$, entonces $G \equiv F$
  - Transitiva: Si $F \equiv G$ y $G \equiv H$, entonces $F \equiv H$

- Principio de sustitución de fórmulas equivalentes:

  ☞ ***Propiedad***. Si en la fórmula F_1 se sustituye una de sus subfórmulas G_1 por una fórmula G_2 lógicamente equivalente a G_1 , entonces la fórmula obtenida, F_2 , es lógicamente equivalente a F_1 .

Ejemplo 21

- $F_1 = \forall x P(x) \to \exists x Q(x)$
- $G_1 = \forall x P(x)$
- $G_2 = \forall y P(y)$
- $F_2 = \forall y P(y) \to \exists x Q(x)$

5 Formas normales

5.1 Forma normal conjuntiva

5.1.1 Definición de forma normal conjuntiva

Definición

Un átomo es una variable proposicional (ejemplo: $p, q, \dots$).

Definición

Un literal es un átomo o su negación (ejemplo: $p, \neg p, q, \neg q, \dots$).

Notación: $L, L_1, L_2, \dots$ representarán literales.

Definición

Una fórmula está en forma normal conjuntiva (FNC) si es una conjunción de disyunciones de literales; es decir, es de la forma $(L_{1,1} \vee \cdots \vee L_{1,n_1}) \wedge \cdots \wedge (L_{m,1} \vee \cdots \vee L_{m,n_m})$.

Ejemplo 1

- $(\neg p \vee q) \wedge (\neg q \vee p)$ está en FNC.
- $(\neg p \vee q) \wedge (q \to p)$ no está en FNC.

Definición

Una fórmula G es una forma normal conjuntiva (FNC) de la fórmula F si G está en forma normal conjuntiva y es equivalente a F.

Ejemplo 2

- Una FNC de $\neg(p \wedge (q \to r))$ es $(\neg p \vee q) \wedge (\neg p \vee \neg r)$.

5.2 Forma normal disyuntiva

5.2.1 Definición de forma normal disyuntiva

Definición

Una fórmula está en forma normal disyuntiva (FND) si es una disyunción de conjunciones de literales; es decir, es de la forma

$$(L_{1,1} \wedge \cdots \wedge L_{1,n_1}) \vee \cdots \vee (L_{m,1} \wedge \cdots \wedge L_{m,n_m}).$$

Ejemplo 7

- $(\neg p \wedge q) \vee (\neg q \wedge p)$ está en FND.
- $(\neg p \wedge q) \vee (q \to p)$ no está en FND.

Definición

Una fórmula G es una forma normal disyuntiva (FND) de la fórmula F si G está en forma normal disyuntiva y es equivalente a F.

Ejemplo 8

Una FND de $\neg(p \wedge (q \to r))$ es $\neg p \vee (q \wedge \neg r)$.

Bibliografía

[1] Calixto Badesa, Ramón Jansana Ferrer, and Ignacio Jané, *Elementos de lógica formal*, Ariel, 1998.

[2] ML Bonet, *Apuntes de lpo*, 2003.

[3] José Luis Fernández, A Manjarrés, and Francisco Javier Díez, *Lógica computacional*, 2003.

[4] Jean H Gallier, *Logic for computer science: foundations of automatic theorem proving*, Courier Dover Publications, 2015.

[5] Michael Huth and Mark Ryan, *Logic in computer science: Modelling and reasoning about systems*, Cambridge university press, 2004.

[6] José A Alonso Jiménez and María J Hidalgo Doblado, *Lógica matemática y fundamentos (2011–12).*