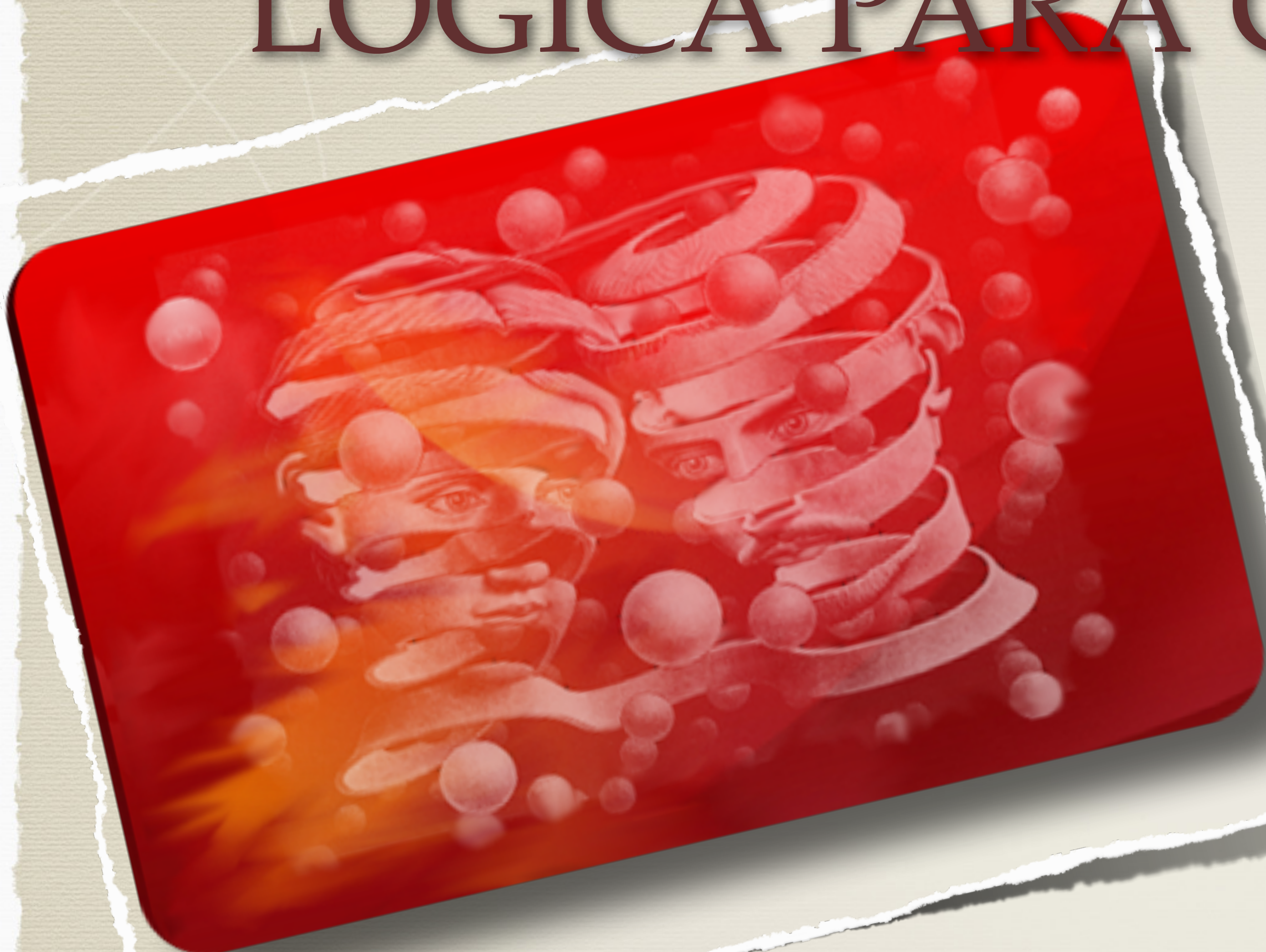
LÓGICA PARA COMPUTACIÓN



LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN

UNSL

Plan 1/23

Lógica para Computación

UNSL

⌘ Equipo de cátedra

- Teorías: Verónica Ludueña
- Prácticos: Cecilia Sosa Toranzo
  Ana Joffre

⌘ Lugar de trabajo

- Oficina: 7 - 1er. piso - Bloque II
- Mails de consultas: logicapc@unsl.edu.ar,
  vlud@unsl.edu.ar,
  ceciliasosatoranzo@gmail.com,
  anjofre@email.unsl.edu.com

Lógica para Computación

UNSL

⌘ Sitio de la asignatura:

http://logica.dirinfo.unsl.edu.ar/

Acá subimos toda la info necesaria para el cursado:

- Apuntes.
- Prácticos.
- Cronograma.
- Horarios.
- Novedades.

Inscribite !!!

Lógica para Computación

UNSL

⌘ Horarios

Lunes

Hora: 09 -11 hs
Aula: 53 - Bloque II

Miércoles

Hora: 14 - 16 hs
Aula: 54 - Bloque II

Lunes

Hora: 08 -10 hs
Aula: Sala 8 - Bloque II

Sin Auxiliares

Lógica para Computación

UNSL

⌘ Correlatividades

Para Cursar

- Algebra I: REGULAR
- Programación I: REGULAR

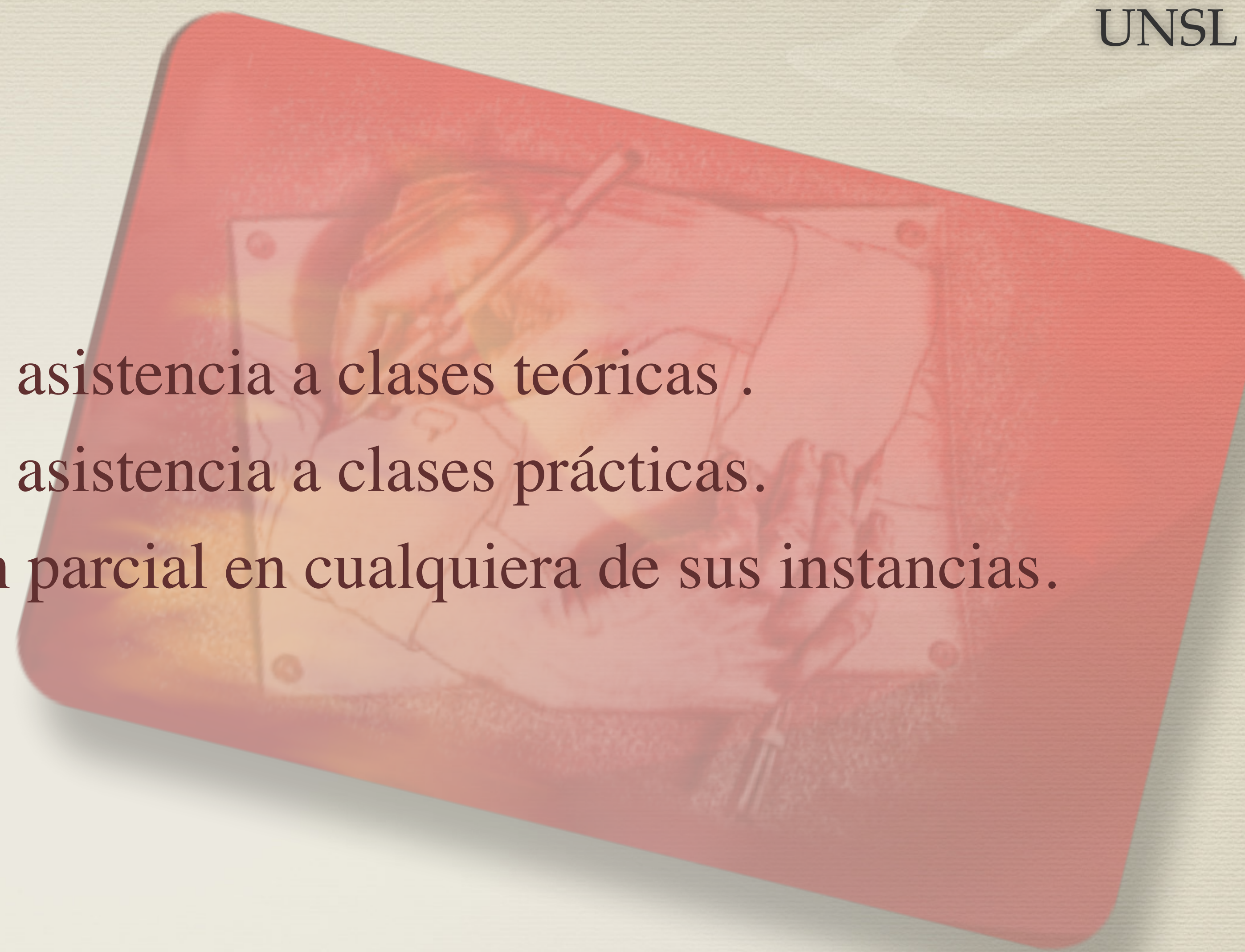
Para Rendir o Promocionar

- No necesita correlativas

Lógica para Computación

UNSL

⌘ Régimen de aprobación



Regularización

- Contar con el 70 % de asistencia a clases teóricas .
- Contar con el 70 % de asistencia a clases prácticas.
- Aprobar la evaluación parcial en cualquiera de sus instancias.

Para Aprobar

- Por promoción:
    - Regularizar con 80% de asistencia y nota 7 o mayor .
    - Rendir un teórico integrador.
- Por examen final:
    - Regularizar y rendir un examen final.

No admite examen libre

Lógica para Computación

UNSL

⌘ Fechas Importantes

Parcial:

Semana del 02 junio

Recuperación:

Semana del 09 junio

2da. Recuperación:

Semana del 23 junio

Teórico: A coordinar

Lógica para Computación

UNSL

Temario

- LÓGICA PROPOSICIONAL
  - ASPECTOS SINTÁCTICOS
  - ASPECTOS SEMÁNTICOS
  - SISTEMA DEDUCTIVO
- LÓGICA de PREDICADOS
  - ASPECTOS SINTÁCTICOS
  - ASPECTOS SEMÁNTICOS
  - DEDUCCIÓN
- DEMOSTRACIÓN MATEMÁTICA
  - MÉTODOS de DEMOSTRACIÓN
- PROGRAMACIÓN LÓGICA
  - PROLOG
- VERIFICACIÓN DE PROGRAMAS

LÓGICA PARA COMPUTACIÓN

LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN

LÓGICA PROPOSICIONAL

ASPECTOS SINTÁCTICOS Y SEMÁNTICOS

UNSL

Lógica para Computación

- La lógica juega un papel básico en la informática, brinda una herramienta teórica para especificaciones formales en diferentes áreas:

    - Lenguajes de programación
    - Diseño de sistemas hardware
    - Diseño y verificación de sistemas software
    - Bases de datos
    - Complejidad computacional
    - Inteligencia artificial

Lógica para Computación

- El objeto de estudio de la lógica son las formas de razonamiento. Tiene como objetivo determinar si nuestros razonamientos son correctos o incorrectos, independientemente de su contenido.

- El razonamiento es el proceso por el cual se derivan conclusiones a partir de premisas y permite justificar ciertas verdades.

- La lógica provee reglas y técnicas para determinar si un argumento dado es válido.

Lógica para Computación

- Para examinar los mecanismos de razonamiento de manera precisa, es necesario utilizar un lenguaje que no dé lugar a confusiones.

- Esto se consigue mediante un lenguaje simbólico en el cual cada símbolo tenga un significado bien definido.

- Uno de estos lenguajes es la *lógica proposicional* o cálculo proposicional, también conocida como lógica de enunciados o lógica de orden cero.

Lógica Proposicional

⌘ La *lógica proposicional* o *cálculo proposicional* es un sistema formal cuyos elementos más simples representan *proposiciones*, y cuyos operadores son *conectivos*.

Proposición → Frase que puede se verdadera o falsa

Oración declarativa (gramática)

Ejemplos:

Mi perro es lanudo Sí

Los nacidos un lunes tienen el pelo lila Sí

¡Hola! No

Base por altura sobre dos No

¿Comemos? No

Sin Ambigüedad

Lógica Proposicional

Ejemplos:

Está lloviendo afuera — Sí

Afuera está lloviendo — Sí

Enuncian la misma proposición

Lo mismo pasa con:

Hoy llueve

Il pleut aujourd'hui

Today it rains

My rabbit is small

mi conejo es pequeño

Lógica Proposicional

La medicina es una ciencia fáctica

Los nacidos un lunes tienen el pelo lila

Los erizos son aves

Proposiciones Simples

Mi perro es lanudo y mi gato blanco

Está nublado o hay sol

Si ayer llovió entonces hoy llueve

No hay personas verdes al oeste de China

Proposiciones Compuestas

Lógica Proposicional

- Las proposiciones compuestas se obtienen a partir de proposiciones simples unidas por conectivos.

Mi perro es lanudo y mi gato blanco	**y**
Está nublado o hay sol	**o**
Si ayer llovió entonces hoy llueve	**Si…entonces**
No hay personas verdes al oeste de China	**no**

- Veamos algunos ejemplos:

Calisto es una luna de Júpiter o Calisto es una luna de Saturno o Calisto no es una luna

Está lloviendo, París es la capital de Francia y no hay personas verdes, o bien Calisto no es una luna, y todo esta bien si hay chocolate.

Lógica Proposicional

- El valor de verdad de las proposiciones simples depende solo de la realidad, de si el hecho o situación que ésta describe es verdad o no.

Las flores son plantas	verdadero
Los erizos son aves	falso

- El valor de verdad de las proposiciones compuestas depende del valor de verdad de las proposiciones simples que la componen y el conectivo que las relaciona.

Las flores son plantas y los erizos son aves	falso
Las flores son plantas o los erizos son aves	verdadero

Lógica Proposicional

- Más ejemplos:

Mi perro es lanudo y los nacidos un lunes tienen el pelo lila

falso

Si mi perro es lanudo entonces los nacidos un lunes tienen el pelo lila

falso

El 2 es divisor de cuatro y es un número par, pero el 3 es impar y el dos es un número par

verdadero

No es cierto que los nacidos un lunes tienen el pelo lila

verdadero

París es la capital de Francia y tiene cerca de dos millones de habitantes

verdadero

Lógica Proposicional

⌘ Por convención las proposiciones simples se representan con letras minúsculas.

La medicina es una ciencia fáctica

p

Pablo es jefe de bomberos y trapecista o Pablo es jefe de bomberos y panadero

p **q** **p** **r**

Está lloviendo, París está en Francia y no hay personas verdes, o bien

p **q** **r**

Calisto no es una luna, y todo esta bien si hay chocolate.

s **t** **u**

Alexia limpia su cuarto y su gato lo llena de pelos; la madre de Alexia se enoja

q **s** **r**

si su gato llena de pelos el cuarto

s

Lógica Proposicional

Sintaxis

- Se traducen las proposiciones del lenguaje natural a un lenguaje simbólico.

Mi perro es lanudo pero mi gato blanco $\quad (p \land q)$

p $\land$ **q**

Los lápices de mi hermana son rojos o negros $\quad (q \lor r)$

q $\lor$ **r**

p = mi perro es lanudo

q = mi gato blanco

q = Los lápices de mi hermana son rojos

r = Los lápices de mi hermana son negros

Lógica Proposicional

⌘ Se traducen las proposiciones del lenguaje natural a un lenguaje simbólico.

Mi perro es lanudo pero mi gato blanco $(p \wedge q)$

Los lápices de mi hermana son rojos o negros $(q \vee r)$

Si ayer llovió , hoy llueve $(p \rightarrow r)$

$p \quad \rightarrow \quad r$

Es falso que todos los unicornios son azules $\neg s$

s

p = ayer llovió

r = hoy llueve ¬

s = todos los unicornios son azules

Lógica Proposicional

- Se traducen las proposiciones del lenguaje natural a un lenguaje simbólico.

Mi perro es lanudo pero mi gato blanco	$(p \wedge q)$
Los lápices de mi hermana son rojos o negros	$(q \vee r)$
Si ayer llovió , hoy llueve	$(p \rightarrow r)$
Es falso que todos los unicornios son azules	$\neg s$

- Esta traducción permite resolver ambigüedades del lenguaje natural y analizar las proposiciones aplicando las herramientas de la lógica.

⌘ Más ejemplos:

A pesar de que $\underbrace{\text{el coche no aceleró}}_{p}$, $\underbrace{\text{hubo un accidente.}}_{q}$ $(p \wedge q)$

$\underbrace{\text{Juan canta}}_{q}$ sólo si $\underbrace{\text{está contento}}_{r}$ $(r \rightarrow q)$

$\underbrace{\text{El asesino ha abandonado el país}}_{p}$ o, $\underbrace{\text{en caso contrario,}}_{r}$ $\underbrace{\text{alguien está encubriéndole}}_{s}$

$(p \vee (r \wedge s))$ O $(p \vee (\neg p \wedge s))$

$\underbrace{\text{María estudiará durante todo el verano}}_{p}$ si $\underbrace{\text{no aprueba lógica}}_{q}$ $\underbrace{\text{ni programación.}}_{r}$

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

Lógica Proposicional

⌘ Veamos algunos términos de enlace del lenguaje natural.

- Conjunción:

  y, pero, aunque, sin embargo, …. $\wedge$

- Disyunción:

  o, o bien, a menos que, …. $\vee$

- Negación:

  no, no es cierto que, es falso que, … $\neg$

- Condicional:

  si…..entonces….., sólo si, ……. $\rightarrow$

- Sea A un conjunto finito y no vacío, denominado ***alfabeto***, indicaremos con A^*, al conjunto de todas las listas finitas de elementos de A.

- Llamaremos ***Lenguaje*** sobre el alfabeto A a cualquier ***subconjunto de A^****.

$$\mathcal{L} \subseteq A^*$$

- Sea $\mathcal{L}$ un lenguaje sobre A, entonces:
  - La lista vacía será indicada por λ.
  - Los elementos del lenguaje son llamados ***palabras***.

Ejemplos:

- $A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

$$\mathcal{L} = A^* - \{\lambda\}$$

Palabras: 0, 12, 12371237438957, 00000057845, 57845, 03, 003, 00003...

- $A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

$$\mathcal{L}' = \mathcal{L} - \{\textit{listas que comienzan con 0}\} \cup \{0\}$$

Palabras: 12, 12439874, 57845, 578450, 3...

- A = Alfabeto castellano $\cup$ {vocales acentuadas}

$$\mathcal{L}'' = \textit{todas las listas que figuran en el diccionario de la RAE.}$$

Palabras: pala, bordar, estudiar,…..

Lógica Proposicional

- Llamaremos ***alfabeto de la lógica proposicional*** al conjunto $\boldsymbol{A}$ tal que:

$$A = \{\, p, \mid, \neg\,, \wedge, \vee, \rightarrow, (\,,\,) \,\}$$

- Llamaremos a los símbolos $\neg, \wedge, \vee, \rightarrow$ ***Conectivos Proposicionales***.

- Las listas $s \in A^*$ formadas por p seguida de un número finito de barras $\mid$ se llamarán ***Variables Proposicionales***.

Ejemplos:

$$p\,,\;\; p\mid,\;\; p\mid\mid\mid,\, p\mid\mid\mid\mid\mid$$

⌘ Para simplificar la escritura, se representará con $\boldsymbol{p_n}$ la variable $\boldsymbol{p}$ seguida de $\boldsymbol{n}$ barras.

Ejemplos:

		Otros sistemas
p	p_0	$\boldsymbol{p}$
$p\,\vert$	p_1	$\boldsymbol{q}$
$p\,\vert\vert\vert$	p_3	$\boldsymbol{r}$

⌘ El subconjunto de $\boldsymbol{A^*}$ que contiene ***todas*** las variables proposicionales será denotado por ***Var***

Lógica Proposicional

⌘ Definimos el ***Lenguaje*** $\mathcal{L}$ de la Lógica Proposicional y lo denotamos por ***Form*** $\subseteq \mathbf{A}^*$, como el conjunto de *fórmulas*, tal que:

⌘ Una lista de símbolos de ***A*** es una ***fórmula*** si sólo si se la puede obtener aplicando un número *finito* de veces las siguientes reglas:

FP1) Las ***variables proposicionales*** son *fórmulas*.

FP2) Si $\boldsymbol{P} \in A^*$ es una *fórmula*, entonces $\neg \boldsymbol{P}$ es una *fórmula*.

FP3) Si $\boldsymbol{P}, \boldsymbol{Q} \in A^*$ son *fórmulas*, entonces

$$(\boldsymbol{P} \wedge \boldsymbol{Q}),\ (\boldsymbol{P} \vee \boldsymbol{Q}),\ (\boldsymbol{P} \to \boldsymbol{Q}) \text{ son } \textit{fórmulas}.$$

FP4) Nada más es *fórmula*.

Lógica Proposicional

Ejemplos de fórmulas:

- ⌘ $\underbrace{p_1}_{P}$ $\qquad$ $\underbrace{p_{123}}_{Q}$ $\qquad$ $\underbrace{p_{34}}_{R}$

- ⌘ $\neg\, p_1 = S$

  $$\underbrace{\neg\, \underbrace{p_1}_{P}}_{S}$$

  $\neg\, (p_{123} \wedge p_{34}) = W$

  $$\underbrace{\neg\, \underbrace{(\underbrace{p_{123}}_{Q} \wedge \underbrace{p_{34}}_{R})}_{U}}_{W}$$

- ⌘ $(p_{123} \rightarrow p_{34}) = T$

  $$\underbrace{(\underbrace{p_{123}}_{P} \rightarrow \underbrace{p_{34}}_{Q})}_{T}$$

  $((p_{13} \vee p_{41}) \rightarrow p_1) = Z$

  $$\underbrace{(\underbrace{(\underbrace{p_{13}}_{V} \vee \underbrace{p_{41}}_{X})}_{O} \rightarrow \underbrace{p_1}_{P})}_{Z}$$

Ejemplos de fórmulas:

- $(\ p_1)$ NO $\quad p_{123}\neg$ NO $\quad (\neg\ p_1)$ NO

  $p_1 \quad \neg\ p_{123} \quad \neg\ p_1$

- $(p_{13} \vee p_{41} \ \vee\ p_1\ )$ $\quad \neg\ (p_{123} \wedge p_{34}$

  $((p_{13} \vee p_{41}\ ) \vee\ p_1\ )$ $\quad \neg\ (p_{123} \wedge p_{34}\ )$

  $(p_{13} \vee (p_{41}\ \vee\ p_1\ ))$

- $p_{123} \rightarrow\ p_{34}$ $\quad ((p_{13} \vee p_{41}\ \rightarrow\ p_1\ )$

  $(\ p_{123} \rightarrow\ p_{34}\ )$ $\quad ((p_{13} \vee p_{41}\ ) \rightarrow\ p_1\ )$

⌘ ***Teorema 2:* (unicidad de lectura)**

Sean P, Q, R y S fórmulas, y supongamos que ★ y ☀ representan algunos de los conectivos ∧, ∨ y →, entonces

$$\textit{Si}\ \ (P \star Q) = (R \,☀\, S)\ \ \textit{entonces}\ \ \underline{P = R},\ \ \underline{Q = S}\ \ \textit{y}\ \ \underline{\star = ☀}$$

⌘ Lo que nos dice este teorema es que cualquiera sea la fórmula que consideremos, no existe más que una única forma de escribirla; se trata del principio de lectura única o unicidad de lectura.

⌘ Cada fórmula compuesta tiene un único conectivo primario y únicas sub fórmulas inmediatas.

$$P = (Q \to R)$$

Existe una única formula Q y una única formula R, que unidas por el → forman la fórmula P.

PREGUNTAS .....

Aspecto Sintáctico

Proposición $\xrightarrow{\textit{formalización}}$ Fórmula

Valor de Verdad

Aspecto Semántico

Verdadero / Falso $\longrightarrow$ Significado $\vee, \wedge, \rightarrow, \neg$

$\top$ $\perp$

V / 1

F / 0

⌘ Interpretación de los Conectivos Proposicionales

"negación"

$\neg$	
$\top$	$\bot$
$\bot$	$\top$

$\sim$ $-$

$\neg(\top)$ es $\bot$ y $\neg(\bot)$ es $\top$

⌘ Interpretación de los Conectivos Proposicionales

"y" , "and"

$\wedge$	$\bot$	$\top$
$\top$	$\bot$	$\top$
$\bot$	$\bot$	$\bot$

& •

$\wedge$ $(\top, \top)$ es $\top$, en cualquier otro caso es $\bot$

⌘ Interpretación de los Conectivos Proposicionales

"o" , "or"

$\vee$	$\perp$	$\top$
$\top$	$\top$	$\top$
$\perp$	$\perp$	$\top$

inclusivo

$\vee\,(\perp , \perp)$ es $\perp$, en cualquier otro caso es $\top$

⌘ Interpretación de los Conectivos Proposicionales

"si ... entonces..", "if ... then..." , implicación

$\rightarrow$	$\perp$	$\top$
$\top$	$\perp$	$\top$
$\perp$	$\top$	$\top$

$\supset$

$\rightarrow$ ($\top$, $\perp$) es $\perp$, en cualquier otro caso es $\top$

Ejemplos:

$(P \wedge Q)$ Si suponemos a P *verdadero* y a Q *falso*

$\top \quad \bot$

$\boxed{\bot}$

$((P \wedge Q)) \rightarrow \neg R)$ Si suponemos a R *falso*

$\top \quad \bot \qquad \bot$

$\bot \qquad \top$

$\boxed{\top}$

⌘ Llamaremos **B** = {⊥,T} y ***Var*** al conjunto de variables proposicionales.

⌘ Una ***valuación booleana*** o, simplemente ***valuación***, es cualquier función v: **Form**$\longmapsto$**B** tal que siendo P y Q fórmulas arbitrarias, satisfaga las siguientes condiciones:

$$(\mathrm{v}_1) \quad v(\neg P) = \neg\, v(P)$$

$$(\mathrm{v}_2) \quad v((P \wedge Q)) = v(P) \wedge v(\mathrm{Q})$$

$$(\mathrm{v}_3) \quad v((P \vee Q)) = v(P) \vee v(\mathrm{Q})$$

$$(\mathrm{v}_4) \quad v((P \rightarrow Q)) = v(P) \rightarrow v(\mathrm{Q})$$

⌘ Decimos que una fórmula P es:

Tautología: si $v\,(P) = \top$.   para *toda* valuación v

Contradicción* o *Falsedad: si $v\,(P) = \bot$ para *toda* valuación v

Contingencia: si P no es ni tautología ni contradicción

Ejemplos: $(p_1 \to p_1)$ ***tautología***

contingencia

P es una *contradicción* si solo si $\neg P$ es una *tautología*

contradicción

$(((p_1 \to p_1) \to (p_2 \vee \neg\, p_2)) \to (p_1 \to p_1)))$ ***tautología***

⌘ Llamaremos *var* (P) al conjunto de variables proposicionales que aparecen en P.

Ejemplos:

$P = (p_3 \wedge \neg (p_4 \rightarrow p_3))$ $\quad$ $var\,(P) = \{p_3, p_4\}$

$P = ((p_1 \rightarrow p_1) \rightarrow (p_1 \vee \neg\, p_1))$ $\quad$ $var\,(P) = \{p_1\}$

⌘ ***Teorema:***

Sea $\boldsymbol{f} : \boldsymbol{Var} \longmapsto \boldsymbol{B}$ *una función cualquiera. Entonces existe una* ***única*** *valuación* $\boldsymbol{v_f} : \boldsymbol{Form} \longmapsto \boldsymbol{B}$ *que extiende a* $\boldsymbol{f}$*, es decir que:*

$$\boldsymbol{v_f(p_i) = f(p_i)} \text{ para toda variable proposicional } p_i$$

⌘ ***Teorema :***

Sean $x_1, ..., x_n$ que representan n variables proposicionales distintas y sea P una fórmula tal que $var\,(P) \subseteq \{x_1, ..., x_n\}$.

Entonces para toda valuación v, el siguiente procedimiento permite calcular el valor $v\,(P)$, conociendo los valores $v\,(x_1), ..., v\,(x_n)$:

Sustituir en la expresión P cada una de las variables x_i por $v(x_i)$, y evaluar la expresión así obtenida en B, utilizando las tablas que interpretan los conectivos lógicos, obteniéndose de esta manera el valor $v\,(P)$.

Método sistemático y de cantidad de pasos finita para determinar si una fórmula es tautología:

Si listamos *todas* las ***n*-uplas** $(a_1, ..., a_n) \in B^n$ con el valor $v_a(P)$ correspondiente, obtenemos todas la valuaciones de P.

¿de cuántas ***n*-upla** estamos hablando?

- Si tenemos dos variables proposicionales p_0 y p_1
- Si tenemos **tres variables proposicionales** $\boldsymbol{p_0}$, p_1 y p_3

p_0	p_1
$\top$	$\top$
$\bot$	$\bot$

$\top$	$\bot$
$\bot$	$\top$
$\top$	$\top$
$\bot$	$\bot$

2^2

p_0	p_1	p_2
$\top$	$\top$	$\top$
$\bot$	$\bot$	$\bot$

$\top$	$\top$	$\top$
$\bot$	$\bot$	$\bot$
$\top$	$\top$	$\bot$
$\bot$	$\bot$	$\top$
$\top$	$\bot$	$\top$
$\bot$	$\top$	$\bot$
$\bot$	$\top$	$\top$
$\top$	$\bot$	$\bot$

2^3

⌘ Si $var(P) = n$ variables proposicionales, tendremos $\mathbf{2^n}$ n-uplas $(a_1, \ldots, a_n) \in \mathrm{B}^n$ con su correspondiente valor $v_a(P)$ cada una.

Luego podemos armar una tabla de

$$\mathbf{2^n} \text{ filas} \times n+1 \text{ columnas}$$

Para obtener todas las valuaciones de P

Tabla de Verdad de la fórmula P

⌘ Dos valuaciones pueden asignar *diferentes valores de verdad a una fórmula P* solamente si asignan diferentes valores de verdad a alguna de las variables proposicionales que intervienen en la escritura de P.

Ejemplos: $P = (((p_0 \vee p_1) \rightarrow \neg p_3) \wedge p_0)$

$x_1 = p_0 \quad x_2 = p_1 \quad x_3 = p_3$

	x_1	x_2	x_3	
$nupla_1$:	$\top$	$\top$	$\top$	$v(P): \bot$
$nupla_2$:	$\top$	$\top$	$\bot$	$v(P): \top$
$nupla_3$:	$\top$	$\bot$	$\top$	$v(P): \bot$
$nupla_4$:	$\top$	$\bot$	$\bot$	$v(P): \top$
$nupla_5$:	$\bot$	$\top$	$\top$	$v(P): \bot$
$nupla_6$:	$\bot$	$\top$	$\bot$	$v(P): \bot$
$nupla_7$:	$\bot$	$\bot$	$\top$	$v(P): \bot$
$nupla_8$:	$\bot$	$\bot$	$\bot$	$v(P): \bot$

p_0	p_1	p_3	P
$\top$	$\top$	$\top$	$\bot$
$\top$	$\top$	$\bot$	$\top$
$\top$	$\bot$	$\top$	$\bot$
$\top$	$\bot$	$\bot$	$\top$
$\bot$	$\top$	$\top$	$\bot$
$\bot$	$\top$	$\bot$	$\bot$
$\bot$	$\bot$	$\bot$	$\bot$
$\bot$	$\bot$	$\top$	$\bot$

Ejemplos: $P = (((p_0 \vee p_1) \rightarrow \neg p_3) \wedge p_0)$

$x_1 = p_0 \quad x_2 = p_1 \quad x_3 = p_3$

p_0	p_1	p_3	P
$\top$	$\top$	$\top$	$\bot$
$\top$	$\top$	$\bot$	$\top$
$\top$	$\bot$	$\top$	$\bot$
$\top$	$\bot$	$\bot$	$\top$
$\bot$	$\top$	$\top$	$\bot$
$\bot$	$\top$	$\bot$	$\bot$
$\bot$	$\bot$	$\bot$	$\bot$
$\bot$	$\bot$	$\top$	$\bot$

$var\,(\mathrm{P}) = \{p_0, p_1, p_3\}$

Tabla de Verdad de *P*

Para saber si es tautología se mira la última columna

⌘ **Observación**

La *Tabla de Verdad* de una fórmula P , con *var* $(P) = \{x_0, x_1, .. x_n\}$

puede verse como una función:

$$T_P : B^n \rightarrow B$$

$$T_P\,(v\,(x_1), \ldots, v\,(x_n)) = v\,(P)$$

Funciones Booleanas de n variables

- Las fórmulas P y Q se dicen ***equivalentes*** y se denota $\boldsymbol{P \equiv Q}$, si para toda valuación v se tiene $v(P) = v(Q)$.

Propiedades:

- Reflexiva: $P \equiv P$
- Simétrica: si $P \equiv Q$ entonces $Q \equiv P$
- Transitiva: si $P \equiv Q$ y $Q \equiv R$ entonces $P \equiv R$

Ejemplo:

$(P \rightarrow Q)$ es equivalente a $(\neg P \vee Q)$

- Las fórmulas P y Q se dicen ***equivalentes*** y se denota $\boldsymbol{P \equiv Q}$, si para toda valuación v se tiene $v(P) = v(Q)$.

Propiedades:

- Reflexiva: $P \equiv P$
- Simétrica: si $P \equiv Q$ entonces $Q \equiv P$
- Transitiva: si $P \equiv Q$ y $Q \equiv R$ entonces $P \equiv R$

Lema:

Si var (P) = var (Q) entonces $P \equiv Q$ si y solo si P y Q tienen la misma Tabla de Verdad.

⌘ Observación:

La hipótesis $var\ (P) = var\ (Q)$ es esencial para la validez del Lema anterior

Sean $P = (p_1 \vee p_2)$ $\quad Q = (p_1 \vee p_3)$

Si v es una valuación tal que $v\ (p_1) = v\ (p_2) = \bot$ y $v\ (p_3) = \top$ entonces

$$v\ (P) = \bot \text{ y } v\ (Q) = \top$$

Luego P no es equivalente a Q aunque $T_P(a_1, a_2) = T_Q(a_1, a_2)$ para toda $(a_1, a_2) \in B^2$.

⌘ Teorema:

Las fórmulas P y Q son equivalentes si y solo si $(P \to Q)$ y $(Q \to P)$ son ambas tautologías.

⌘ Ejemplos: de algunas equivalencias útiles:

- $\neg (P \wedge Q) \equiv (\neg P \vee \neg Q)$
- $(P \rightarrow Q) \equiv (\neg P \vee Q)$
- $\neg (P \vee Q) \equiv (\neg P \wedge \neg Q)$
- $((P \rightarrow Q) \wedge (Q \rightarrow P)) \equiv (P \leftrightarrow Q)$
- $(P \wedge (Q \wedge R)) \equiv (P \wedge Q) \wedge R))$
- $\neg\neg P \equiv P$
- $(P \vee (Q \vee R)) \equiv (P \vee Q) \vee R))$

Formas Normales

- *Proposición*: Toda función de verdad puede representarse por una fórmula P en la que solo aparecen los conectivos $\wedge$, $\neg$ y $\vee$.

p_1	p_2	p_3	P
$\top$	$\top$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\top$
$\top$	$\bot$	$\top$	$\bot$
$\top$	$\bot$	$\bot$	$\bot$
$\bot$	$\top$	$\top$	$\bot$
$\bot$	$\top$	$\bot$	$\bot$
$\bot$	$\bot$	$\top$	$\bot$
$\bot$	$\bot$	$\bot$	$\top$

$(p_1 \wedge p_2 \wedge p_3)$

$(p_1 \wedge p_2 \wedge \neg p_3)$ *Conjunciones simples*

$(\neg p_1 \wedge \neg p_2 \wedge \neg p_3)$

$$((p_1 \wedge p_2 \wedge p_3) \vee (p_1 \wedge p_2 \wedge \neg p_3) \vee (\neg p_1 \wedge \neg p_2 \wedge \neg p_3))$$

c.q.d.

Forma Normal Disyuntiva

⌘ Toda fórmula P que *no sea una contradicción* es equivalente a una fórmula Q del siguiente estilo:

$$\boxed{\left(\bigvee_{i=1}^{m}\left(\bigwedge_{j=1}^{n} Q_{ij}\right)\right)}$$

- m es la cantidad de n-uplas $a = (a_1, ..., a_n) \in B^n$ tal que $v_a(P) = \top$
- n es la cantidad de variables proposicionales.
- Q_{ij} es una variable proposicional o la negación de una.

Forma Normal Conjuntiva

- Toda fórmula P que *no sea una tautología* es equivalente a una fórmula Q del siguiente estilo:

$$\boxed{\left(\bigwedge_{i=1}^{m}\left(\bigvee_{j=1}^{n} Q_{ij}\right)\right)}$$

- m es la cantidad de n-uplas $a = (a_1, ..., a_n) \in B^n$ tal que $v_a(P) = \perp$
- n es la cantidad de variables proposicionales
- Q_{ij} es una variable proposicional o la negación de una.

⌘ Forma Normal Disyuntiva

Ejemplo:

$$((p_1 \wedge p_2 \wedge p_3) \vee (p_1 \wedge p_2 \wedge \neg p_3) \vee (\neg p_1 \wedge \neg p_2 \wedge \neg p_3))$$

p_1	p_2	p_3	P
$\top$	$\top$	$\top$	$\top$
$\top$	$\top$	$\bot$	$\top$
$\top$	$\bot$	$\top$	$\bot$
$\top$	$\bot$	$\bot$	$\bot$
$\bot$	$\top$	$\top$	$\bot$
$\bot$	$\top$	$\bot$	$\bot$
$\bot$	$\bot$	$\top$	$\bot$
$\bot$	$\bot$	$\bot$	$\top$

$(p_1 \wedge p_2 \wedge p_3)$

$(p_1 \wedge p_2 \wedge \neg p_3)$

$(\neg p_1 \wedge \neg p_2 \wedge \neg p_3)$

¿Por qué no se podría usar la FND si P fuera una *contradicción*?

⌘ Forma Normal Conjuntiva

Formalmente la manera de encontrar una fórmula R equivalente a P que esté en Forma Normal Conjuntiva es:

- Si se sabe que P **no es** *Tautolgía*, entonces $\neg P$ **no es** *Contradicción*.

- Entonces $\neg P$ es lógicamente equivalente a una fórmula en forma normal disyuntiva.

$$\neg P \equiv \left(\vee_{i=1}^{m} \left(\wedge_{j=1}^{n} Q_{ij} \right) \right)$$

- Luego P es lógicamente equivalente a:

$$P \equiv \neg \left(\vee_{i=1}^{m} \left(\wedge_{j=1}^{n} Q_{ij} \right) \right)$$

⌘ Forma Normal Conjuntiva

- Por las leyes de De Morgan, esto:

$$P \equiv \neg\left(\vee_{i=1}^{m}\left(\wedge_{j=1}^{n} Q_{ij}\right)\right)$$

- Es lógicamente equivalente a:

$$P \equiv \left(\wedge_{i=1}^{m}\left(\vee_{j=1}^{n} \neg Q_{ij}\right)\right) = R$$

- Finalmente se reemplaza en R cada aparición de la expresión $\neg\neg q$ por q .

⌘ Forma Normal Conjuntiva

Otra la manera de encontrar una fórmula R equivalente a P que esté en Forma Normal Conjuntiva es:

- Identificamos las filas donde $v(P)$ es falso
- Obtenemos las disyunciones simples.
- Armamos la conjunción.

p_1	p_2	p_3	P	
$\top$	$\top$	$\top$	$\top$	
$\top$	$\top$	$\bot$	$\top$	
$\top$	$\bot$	$\top$	$\bot$	$\longrightarrow (\neg p_1 \vee p_2 \vee \neg p_3)$
$\top$	$\bot$	$\bot$	$\bot$	$\longrightarrow (\neg p_1 \vee p_2 \vee p_3)$
$\bot$	$\top$	$\top$	$\bot$	$\longrightarrow (p_1 \vee \neg p_2 \vee \neg p_3)$
$\bot$	$\top$	$\bot$	$\bot$	$\longrightarrow (p_1 \vee \neg p_2 \vee p_3)$
$\bot$	$\bot$	$\top$	$\bot$	$\longrightarrow (p_1 \vee p_2 \vee \neg p_3)$
$\bot$	$\bot$	$\bot$	$\top$	

¿Por qué no se podría usar la FNC si P fuera una *tautología*?

$$\boldsymbol{((\neg p_1 \vee p_2 \vee \neg p_3) \wedge (\neg p_1 \vee p_2 \vee p_3) \wedge (p_1 \vee \neg p_2 \vee \neg p_3) \wedge (p_1 \vee \neg p_2 \vee p_3) \wedge (p_1 \vee p_2 \vee \neg p_3))}$$

⌘ Conjuntos adecuados de conectivos

- ⌘ Un *conjunto adecuado de conectivos* es un conjunto tal que toda función de verdad puede representarse por medio de una fórmula que contenga ***solamente*** conectivos del conjunto.

- ⌘ Son conjuntos adecuados de conectivos

$$\{\wedge, \neg\}$$

$$\{\vee, \neg\}$$

$$\{\rightarrow, \neg\}$$

PREGUNTAS .....