# Lògica en la Informàtica
# Definició de Lògica Proposicional (LProp)

José Miguel Rivero

Dept. Ciències de la Computació
Universitat Politècnica de Catalunya (UPC)

Primavera 2025

# Crèdits

El material utilitzat en aquesta presentació ha estat extret del elaborat pel professor Robert Nieuwenhuis (Dept. CS, UPC) per l'assignatura *Lògica en la Informàtica* de la FIB.

En particular, del llibre *Lógica para informáticos* - Farré, R. [et al.], Marcombo, 2011. ISBN: 9788426716941.

## Temari

# Definició de Lògica Proposicional

Exercicis del capítol 2 dels apunts: ☞ `p2.pdf`

1. Exercici 5 [ demostració en LProp ]
2. Exercici 7 [ demostració en LProp ]
3. Exercici 8 [ demostració en LProp ]
4. Exercici 16 [ demostració en LProp ]
5. Exercici 21 [ equivalència lògica ]
6. Exercici 18 [ equivalències entre fòrmules ]
7. Exercici 23 [ lema de Substitució ]

# Exercici 5

5. (dificultat 1) Siguin $F$ i $G$ dues fórmules qualssevol. És cert que $F \vee G$ és tautologia si i només si alguna de les dues fórmules $F$ o $G$ ho és?. Demostra-ho fent servir només la definició de la LProp.

No, no és cert. Donarem un **contraexemple** de fórmules $F$ i $G$ per a les quals la propietat és falsa, és a dir, on $F \vee G$ és tautologia, però ni $F$ ni $G$ són tautologies.

Sigui $F$ la fórmula $p$, on $p$ és un símbol de predicat.
Sigui $G$ la fórmula $\neg p$.

# Exercici 5

5. (dificultat 1) Siguin $F$ i $G$ dues fórmules qualssevol. És cert que $F \vee G$ és tautologia si i només si alguna de les dues fórmules $F$ o $G$ ho és?. Demostra-ho fent servir només la definició de la LProp.

a) Tenim que $p \vee \neg p$ és tautologia perquè:

| | | |
|---|---|---|
| $p \vee \neg p$ és tautologia | ssi | [ per definició de tautologia ] |
| tota $I$ és model de $p \vee \neg p$ | ssi | [ per definició de model ] |
| $\forall\, I,\ I \models p \vee \neg p$ | ssi | [ per definició de $\models$ ] |
| $\forall\, I,\ eval_I(p \vee \neg p) = 1$ | ssi | [ per definició de $eval_I(\ldots \vee \ldots)$ ] |
| $\forall\, I,\ max(eval_I(p), eval_I(\neg p)) = 1$ | ssi | [ per definició de $eval_I(\neg \ldots)$ ] |
| $\forall\, I,\ max(eval_I(p), 1 - eval_I(p)) = 1$ | ssi | [ donat que $eval_I(p)$ sempre és 0 o 1, i per definició de $max$ ] |

$$max(0, 1-0) = max(1, 1-1) = 1$$

# Exercici 5

5. (dificultat 1) Siguin $F$ i $G$ dues fórmules qualssevol. És cert que $F \vee G$ és tautologia si i només si alguna de les dues fórmules $F$ o $G$ ho és?. Demostra-ho fent servir només la definició de la LProp.

a) Tenim que $p \vee \neg p$ és tautologia perquè:

| | | |
|---|---|---|
| $p \vee \neg p$ és tautologia | ssi | [ per definició de tautologia ] |
| tota $I$ és model de $p \vee \neg p$ | ssi | [ per definició de model ] |
| $\forall I,\ I \models p \vee \neg p$ | ssi | [ per definició de $\models$ ] |
| $\forall I,\ eval_I(p \vee \neg p) = 1$ | ssi | [ per definició de $eval_I(\ldots \vee \ldots)$ ] |
| $\forall I,\ max(eval_I(p), eval_I(\neg p)) = 1$ | ssi | [ per definició de $eval_I(\neg \ldots)$ ] |
| $\forall I,\ max(eval_I(p), 1 - eval_I(p)) = 1$ | ssi | [ donat que $eval_I(p)$ sempre és 0 o 1, i per definició de *max* ] |
| $1 = 1$ | ssi | [ perquè $1 = 1$ és *cert* ] |
| cert | | |

# Exercici 5

5. (dificultat 1) Siguin $F$ i $G$ dues fórmules qualssevol. És cert que $F \vee G$ és tautologia si i només si alguna de les dues fórmules $F$ o $G$ ho és?. Demostra-ho fent servir només la definició de la LProp.

b) Però tenim que ni $F$ ni $G$ són tautologies:

| | | |
|---|---|---|
| $F$ no és tautologia | ssi | [ com $F$ és $p$ ] |
| $p$ no és tautologia | ssi | [ per definició de tautologia ] |
| $\exists\, I$, tq $I$ no és model de $p$ | ssi | [ per definició de model ] |
| $\exists\, I$, $I \not\models p$ | ssi | [ per definició de $\models$ ] |
| $\exists\, I$, $eval_I(p) = 0$ | ssi | [ per definició de $eval_I(p)$ ] |
| $\exists\, I$, $I(p) = 0$ | ssi | cert |

# Exercici 5

5. (dificultat 1) Siguin $F$ i $G$ dues fórmules qualssevol. És cert que $F \vee G$ és tautologia si i només si alguna de les dues fórmules $F$ o $G$ ho és?. Demostra-ho fent servir només la definició de la LProp.

b) Però tenim que ni $F$ ni $G$ són tautologies:

| | | |
|---|---|---|
| $G$ no és tautologia | ssi | [ com $G$ és $\neg p$ ] |
| $\neg p$ no és tautologia | ssi | [ per definició de tautologia ] |
| $\exists\, I$, tq $I$ no és model de $\neg p$ | ssi | [ per definició de model ] |
| $\exists\, I,\ I \not\models \neg p$ | ssi | [ per definició de $\models$ ] |
| $\exists\, I,\ eval_I(\neg p) = 0$ | ssi | [ per definició de $eval_I(\neg \ldots)$ ] |
| $\exists\, I,\ 1 - eval_I(p) = 0$ | ssi | [ per definició de $eval_I(p)$ ] |
| $\exists\, I,\ 1 - I(p) = 0$ | ssi | [ per aritmètica ] |
| $\exists\, I,\ I(p) = 1$ | ssi | cert |

# Exercici 7

7. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Demostra, fent servir tan sols la definició de la LProp, que $F$ és conseqüència lògica de $G$, és a dir, $G \models F$ si i només si $G \wedge \neg F$ és insatisfactible.

| | | |
|---|---|---|
| $F$ és conseqüència lògica de $G$ | ssi | [ per def. de conseqüència lògica ] |
| tot model de $G$ satisfà $F$ | ssi | [ per def. de model ] |
| $\forall I$, si $I \models G$, llavors $I \models F$ | ssi | [ pel significat de si. . . llavors. . . ] |
| $\forall I$, $I \not\models G$ o bé $I \models F$ | ssi | [ per def. de $\models$ ] |
| $\forall I$, $eval_I(G) = 0$ o bé $eval_I(F) = 1$ | | |

$G \wedge \neg F$ és insatisfactible

# Exercici 7

7. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Demostra, fent servir tan sols la definició de la LProp, que $F$ és conseqüència lògica de $G$, és a dir, $G \models F$ si i només si $G \wedge \neg F$ és insatisfactible.

| | | |
|---|---|---|
| $F$ és conseqüència lògica de $G$ | ssi | [ per def. de conseqüència lògica ] |
| tot model de $G$ satisfà $F$ | ssi | [ per def. de model ] |
| $\forall I$, si $I \models G$, llavors $I \models F$ | ssi | [ pel significat de si. . . llavors. . . ] |
| $\forall I$, $I \not\models G$ o bé $I \models F$ | ssi | [ per def. de $\models$ ] |
| $\forall I$, $eval_I(G) = 0$ o bé $eval_I(F) = 1$ | ssi | [ per aritmètica ] |
| $\forall I$, $eval_I(G) = 0$ o bé $1 - eval_I(F) = 0$ | ssi | [ per def. de *min* ] |
| $\forall I$, $min(eval_I(G), 1 - eval_I(F)) = 0$ | ssi | [ per def. de $eval_I(\neg \ldots)$ ] |
| $\forall I$, $min(eval_I(G), eval_I(\neg F)) = 0$ | ssi | [ per def. de $eval_I(\ldots \wedge \ldots)$ ] |
| $\forall I$, $eval_I(G \wedge \neg F) = 0$ | ssi | [ per def. de $\models$ ] |
| $\forall I$, $I \not\models (G \wedge \neg F)$ | ssi | [ per def. de model ] |
| $G \wedge \neg F$ no té models | ssi | [ per def. de insatisfactible ] |
| $G \wedge \neg F$ és insatisfactible | | |

# Exercici 8

8. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Demostra, fent servir tan sols la definició de la LProp, que $F$ és lògicament equivalent a $G$ ssi $(G \wedge \neg F) \vee (F \wedge \neg G)$ és insatisfactible ssi $F \leftrightarrow G$ és tautologia.

Fem primer el primer ssi:

$(G \wedge \neg F) \vee (F \wedge \neg G)$ és insatisfactible

ssi [ per definició de insatisfactible ]

$\forall I,\ I \not\models (G \wedge \neg F) \vee (F \wedge \neg G)$

ssi [ per definició de $\models$ ]

$\forall I,\ eval_I((G \wedge \neg F) \vee (F \wedge \neg G)) = 0$

ssi [ per definició de *eval* $\vee$ ]

$\forall I,\ max(eval_I(G \wedge \neg F), eval_I(F \wedge \neg G)) = 0$

ssi [ per definició de *eval* $\wedge$ ]

$\forall I,\ max(min(eval_I(G), eval_I(\neg F)), min(eval_I(F), eval_I(\neg G))) = 0$

# Exercici 8

8. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Demostra, fent servir tan sols la definició de la LProp, que $F$ és lògicament equivalent a $G$ ssi $(G \wedge \neg F) \vee (F \wedge \neg G)$ és insatisfactible ssi $F \leftrightarrow G$ és tautologia.

Fem primer el primer ssi (cont.):

$\forall I,\ max(min(eval_I(G), eval_I(\neg F)), min(eval_I(F), eval_I(\neg G))) = 0$

ssi [ per definició de $eval$ $\neg$ ]

$\forall I,\ max(min(eval_I(G), 1 - eval_I(F)), min(eval_I(F), 1 - eval_I(G))) = 0$

ssi [ per definició de $min$ i $max$, i perquè $eval$ sempre dona 0 o 1 ]

$\forall I,\ eval_I(F) = eval_I(G)$

ssi [ perquè $eval$ sempre dona 0 o 1 ]

$\forall I,\ (eval_I(F) = 1 \text{ ssi } eval_I(G) = 1)$

ssi [ per definició de $\models$ ]

$\forall I,\ (I \models F \text{ ssi } I \models G)$

# Exercici 8

8. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Demostra, fent servir tan sols la definició de la LProp, que $F$ és lògicament equivalent a $G$ ssi $(G \wedge \neg F) \vee (F \wedge \neg G)$ és insatisfactible ssi $F \leftrightarrow G$ és tautologia.

Fem primer el primer ssi (cont.):

$\forall I, (I \models F$ ssi $I \models G)$
ssi [ per definició de model ]
$\forall I, (I$ és model de $F$ ssi $I$ és model de $G)$
ssi [ per definició de equivalència de models ]
$F$ i $G$ tenen els mateixos models
ssi [ per definició de equivalència lògica ]
$F$ és lógicamente equivalente a $G$.

# Exercici 8

8. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Demostra, fent servir tan sols la definició de la LProp, que $F$ és lògicament equivalent a $G$ ssi $(G \land \neg F) \lor (F \land \neg G)$ és insatisfactible ssi $F \leftrightarrow G$ és tautologia.

Per al segon ssi:

$F \leftrightarrow G$ és tautologia

ssi [ per def. $\leftrightarrow$ ]

$(F \to G) \land (G \to F)$ és tautologia

ssi [ per def. $\to$ ]

$(\neg F \lor G) \land (\neg G \lor F)$ és tautologia

ssi [ per def. de tautologia ]

$\forall I$ és model de $(\neg F \lor G) \land (\neg G \lor F)$

ssi [ per def. de model ]

$\forall I, I \models (\neg F \lor G) \land (\neg G \lor F)$

ssi [ per def. de $\models$ ]

$\forall I, eval_I( (\neg F \lor G) \land (\neg G \lor F) ) = 1$

# Exercici 8

8. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Demostra, fent servir tan sols la definició de la LProp, que $F$ és lògicament equivalent a $G$ ssi $(G \wedge \neg F) \vee (F \wedge \neg G)$ és insatisfactible ssi $F \leftrightarrow G$ és tautologia.

Per al segon ssi (cont.):

$\forall\, I,\ eval_I(\ (\neg F \vee G) \wedge (\neg G \vee F)\ ) = 1$

ssi [ per def. de $eval_I(\ \wedge\ )$ ]

$\forall\, I,\ min(eval_I(\neg F \vee G), eval_I(\neg G \vee F)) = 1$

ssi [ per def. de $eval_I(\ \vee\ )$ ]

$\forall\, I,\ min(max(eval_I(\neg F), eval_I(G)), max(eval_I(\neg G), eval_I(F))) = 1$

ssi [ per def. de $eval_I(\ \neg\ )$ ]

$\forall\, I,\ min(max(1 - eval_I(F), eval_I(G)), max(1 - eval_I(G), eval_I(F))) = 1$

ssi [ per definició de $min$ i $max$, i perquè $eval_I$ sempre dona 0 o 1 ]

$\forall\, I,\ eval_I(F) = eval_I(G)$

i seguim igual que en la demostració anterior.

Conseqüència dels exercicis 6,7,8

En la pràctica ens interessa sempre, esbrinar aquest tipus de propietats:

$I$ **és model** de $F$ si $I$ satisfà a $F$ (es denota $I \models F$)
$F$ és **satisfactible** si $F$ té algun model
$F$ és **insatisfactible** si $F$ no té models
$F$ és **tautologia** si tota $I$ és model de $F$
$G$ és **conseqüència lògica** de $F$ si tot model de $F$ satisfà $G$ (es denota $F \models G$)
$F$ i $G$ són **lògicament equivalents** si $F$ i $G$ tenen el mateixos models (es denota $F \equiv G$)

Com ho podem fer, si l'única cosa que tenim és un SAT solver?

| | | |
|---|---|---|
| $F$ és tautologia | ssi | $\neg F$ és insatisfactible. |
| $G$ és conseqüència lògica de $F$ | ssi | $F \wedge \neg G$ és insatisfactible. |
| $F$ i $G$ són lògicamente equivalentes | ssi | $(G \wedge \neg F) \vee (F \wedge \neg G)$ és insatisfactible. |

# Exercici 16

16. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Si $F \to G$ és satisfactible i $F$ és satisfactible, llavors $G$ és satisfactible? Demostra-ho fent servir tan sols la definició de la LProp.

Farem un *intent de demostració*:

| | | |
|---|---|---|
| $F \to G$ és satisfactible | ssi | [ per def. de $\to$ ] |
| $\neg F \lor G$ és satisfactible | ssi | [ per def. de satisfactible ] |
| $\neg F \lor G$ té algun model | ssi | [ per def. de model ] |
| $\exists\, I,\ I \models \neg F \lor G$ | ssi | [ per def. de $\models$ ] |
| $\exists\, I,\ eval_I(\neg F \lor G) = 1$ | ssi | [ per def. de $eval_I(\ \lor\ )$ ] |
| $\exists\, I,\ max(eval_I(\neg F), eval_I(G)) = 1$ | ssi | [ per def. de $eval_I(\ \neg\ )$ ] |
| $\exists\, I,\ max(1 - eval_I(F), eval_I(G)) = 1$ | ssi | [ per def. de $max$ ] |
| $\exists\, I,\ 1 - eval_I(F) = 1$ o bé $eval_I(G) = 1$ | ssi | [ per aritmètica ] |
| $\exists\, I,\ eval_I(F) = 0$ o bé $eval_I(G) = 1$ | | |

NO puc escriure $\exists\, I,\ eval_I(F) = 0 \lor eval_I(G) = 1$
NO té cap sentit, perquè $\lor$ és una connectiva que només té sentit dintre de fórmules, i $eval_I(F) = 0$ NO és una fórmula. Aquí estem raonant/explicant en català.

16. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Si $F \rightarrow G$ és satisfactible i $F$ és satisfactible, llavors $G$ és satisfactible? Demostra-ho fent servir tan sols la definició de la LProp.

Farem un *intent de demostració (cont)*:

| | | |
|---|---|---|
| $F$ és satisfactible | ssi | [ per def. de satisfactible ] |
| $F$ té algun model | ssi | [ per def. de model ] |
| $\exists I', I' \models F$ | ssi | [ per def. de $\models$ ] |
| $\exists I', eval_{I'}(F) = 1$ | | |

D'aquest intent de demostració, veiem que si existeix una $I$ que no és model de $F$ i un altre $I'$ que sí és model de $F$, llavors ja és compleixen les dues condicions de que $F \rightarrow G$ és satisfactible i $F$ és satisfactible, i això no implica res sobre la $G$!

# Exercici 16

16. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Si $F \to G$ és satisfactible i $F$ és satisfactible, llavors $G$ és satisfactible? Demostra-ho fent servir tan sols la definició de la LProp.

> D'aquest intent de demostració, veiem que si existeix una $I$ que no és model de $F$ i un altre $I'$ que sí és model de $F$, llavors ja és compleixen les dues condicions de que $F \to G$ és satisfactible i $F$ és satisfactible, i això no implica res sobre la $G$!

Això ens inspira per a adonar-nos que la propietat és falsa, i per a donar aquest contraexemple:

Sigui $F$ la fórmula $p$

Sigui $G$ la fórmula $p \wedge \neg p$.

Llavors $F \to G$ és satisfactible i $F$ és satisfactible, però $G$ no ho és!

# Exercici 16

16. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Si $F \to G$ és satisfactible i $F$ és satisfactible, llavors $G$ és satisfactible? Demostra-ho fent servir tan sols la definició de la LProp.

| | | |
|---|---|---|
| $F \to G$ és satisfactible | ssi | [ per def. de $F$ i $G$ ] |
| $p \to (p \wedge \neg p)$ és satisfactible | ssi | [ per def. de $\to$ ] |
| $\neg p \vee (p \wedge \neg p)$ és satisfactible | ssi | [ per def. de satisfactible ] |
| $\neg p \vee (p \wedge \neg p)$ té algun model | ssi | [ per def. de model ] |
| $\exists\, I,\ I \models \neg p \vee (p \wedge \neg p)$ | ssi | [ per def. de $\models$ ] |
| $\exists\, I,\ eval_I(\neg p \vee (p \wedge \neg p)) = 1$ | ssi | [ per def. de $eval_I(\ \vee\ )$ ] |
| $\exists\, I,\ max(eval_I(\neg p), eval_I(p \wedge \neg p)) = 1$ | ssi | [ per def. de $eval_I(\ \neg\ )$ ] |
| $\exists\, I,\ max(1 - eval_I(p), eval_I(p \wedge \neg p)) = 1$ | ssi | [ per def. de $max$ ] |
| $\exists\, I,\ 1 - eval_I(p) = 1$ o $eval_I(p \wedge \neg p) = 1$ | ssi | [ per aritmètica ] |
| $\exists\, I,\ eval_I(p) = 0$ o $eval_I(p \wedge \neg p) = 1$ | ssi | [ agafem la $I$ tal que $I(p) = 0$ ] |

cert

# Exercici 16

16. (dificultat 2) Siguin $F$ i $G$ dues fórmules qualssevol. Si $F \rightarrow G$ és satisfactible i $F$ és satisfactible, llavors $G$ és satisfactible? Demostra-ho fent servir tan sols la definició de la LProp.

| | | |
|---|---|---|
| $F$ és satisfactible | ssi | [ per def. de $F$ ] |
| $p$ és satisfactible | ssi | [ per def. de satisfactible ] |
| $p$ té algun model | ssi | [ per def. de model ] |
| $\exists\, I',\, I' \models p$ | ssi | [ per def. de $\models$ ] |
| $\exists\, I',\, eval_{I'}(p) = 1$ | ssi | [ agafem la $I'$ tal que $I'(p) = 1$ ] |
| cert | | |

$G$ és insatisfactible (☞ veure exercici 2).

# Exercici 21

21. (dificultat 2) Demostra que l'equivalència lògica és realment una relació d'**equivalència**.

Una relacion binària $R$ sobre un conjunt $S$ és un subconjunt del producte cartesià $S \times S$. És a dir $R$ ens diu quines parelles estan relacionades, quines parelles $(e, e')$ estan en $R$ (on $e$ i $e'$ són elements de $S$).

| | | | |
|---|---|---|---|
| $R$ és **reflexiva** si $(e, e)$ està en $R$ | | per a tot $e$ | de $S$. |
| $R$ és **simètrica** si $(e, e')$ en $R$ | implica $(e', e)$ en $R$ | per a tot $e$,$e'$ | de $S$. |
| $R$ és **transitiva** si $(e, e')$ en $R$ i $(e', e'')$ en $R$ | implica $(e, e'')$ en $R$ | per a tot $e$,$e'$,$e''$ | de S. |

I si $R$ compleix les tres propietats llavors $R$ és una relació d'**equivalència**.

# Exercici 21

21. (dificultat 2) Demostra que l'equivalència lògica és realment una relació d'**equivalència**.

Una relacion binària $R$ sobre un conjunt $S$ és un subconjunt del producte cartesià $S \times S$. És a dir $R$ ens diu quines parelles estan relacionades, quines parelles $(e, e')$ estan en $R$ (on $e$ i $e'$ són elements de $S$).

Altres notacions:

- com un predicat binari:

| | | |
|---|---|---|
| $R$ és **reflexiva** si $R(e, e)$ | | per a tot $e$ |
| $R$ és **simètrica** si $R(e, e')$ | implica $R(e', e)$ | per a tot $e$,$e'$ |
| $R$ és **transitiva** si $R(e, e')$ i $R(e', e'')$ | implica $R(e, e'')$ | per a tot $e$,$e'$,$e''$ |

# Exercici 21

21. (dificultat 2) Demostra que l'equivalència lògica és realment una relació d'**equivalència**.

Una relacion binària $R$ sobre un conjunt $S$ és un subconjunt del producte cartesià $S \times S$. És a dir $R$ ens diu quines parelles estan relacionades, quines parelles $(e, e')$ estan en $R$ (on $e$ i $e'$ són elements de $S$).

Altres notacions:

- com un predicat infix:

| | | | |
|---|---|---|---|
| $R$ és **reflexiva** si $eRe$ | | per a tot $e$ | de $S$. |
| $R$ és **simètrica** si $eRe'$ | implica $e'Re$ | per a tot $e$,$e'$ | de $S$. |
| $R$ és **transitiva** si $eRe'$ i $e'Re''$ | implica $eRe''$ | per a tot $e$,$e'$,$e''$ | de S. |

Per exemple si $R$ és $>$, la notació infixa és molt més habitual: escribim $e > e'$, etc.

# Exercici 18

18. (dificultat 2) Demostra les següents equivalències entre fórmules:

| | | | |
|---|---|---|---|
| $F \wedge F$ | $\equiv$ | F | idempotència de $\wedge$ |
| $F \vee F$ | $\equiv$ | F | idempotència de $\vee$ |
| $F \wedge G$ | $\equiv$ | $G \wedge F$ | conmutativitat de $\wedge$ |
| $F \vee G$ | $\equiv$ | $G \vee F$ | conmutativitat de $\vee$ |
| $(F \wedge G) \wedge H$ | $\equiv$ | $F \wedge (G \wedge H)$ | associativitat de $\wedge$ |
| $(F \vee G) \vee H$ | $\equiv$ | $F \vee (G \vee H)$ | associativitat de $\vee$ |

Aquestes tres propietats (idempotència, commutativitat, asociatividad de $\wedge$ i de $\vee$) ens indiquen que a vegades podem escriure les fórmules de manera més "relaxada", ometent alguns parèntesis. I també, que podem veure una CNF com un CONJUNT (and) de clàusules, i podem veure una clàusula com un CONJUNT (un or) de literals.

# Exercici 18

18. (dificultat 2) Demostra les següents equivalèncias entre fórmules:

| | | | |
|---|---|---|---|
| $\neg\neg F$ | $\equiv$ | $F$ | doble negació |
| $\neg(F \wedge G)$ | $\equiv$ | $\neg F \vee \neg G$ | llei de De Morgan 1 |
| $\neg(F \vee G)$ | $\equiv$ | $\neg F \wedge \neg G$ | llei de De Morgan 2 |

Aquestes tres propietats ens serveixen per a transformar fórmules "movent les negacions cap a dins", fins que només hi hagi negacions aplicades a símbols de predicat.

# Exercici 18

18. (dificultat 2) Demostra les següents equivalències entre fórmules:

$$
\begin{array}{lcll}
(F \wedge G) \vee H & \equiv & (F \vee H) \wedge (G \vee H) & \text{distributivitat 1} \\
(F \vee G) \wedge H & \equiv & (F \wedge H) \vee (G \wedge H) & \text{distributivitat 2}
\end{array}
$$

Una vegada les negacions estan aplicades als símbols de predicat, aplicant distributivitat 1 $(F \wedge G) \vee H \Longrightarrow (F \vee H) \wedge (G \vee H)$ d'esquerra a dreta obtenim una CNF.

Hi ha un detall: Demostra que $p \wedge (q \vee q) \equiv p \wedge q$.
Podem "aplicar" alegrement la idempotència del $\vee$ sobre la subfórmula $q \vee q$?
**No**! Cal demostrar primer el Lema de Substitució de l'exercici 23.

# Exercici 18

Demostrem, com a exemple, que $F \equiv \neg\neg F$:

$F \equiv \neg\neg F$

ssi [ per def. d'equivalència lògica ]

$\forall I$, ($I$ és model de $F$ ssi $I$ és model de $\neg\neg F$)

ssi [ per def. de model ]

$\forall I, (I \models F \text{ ssi } I \models \neg\neg F)$

ssi [ per def. de satisfacció ]

$\forall I, (eval_I(F) = 1 \text{ ssi } eval_I(\neg\neg F) = 1)$

ssi [ perquè $eval_I$ sempre dona 0 o 1 ]

$\forall I, eval_I(F) = eval_I(\neg\neg F)$

ssi [ per def. de $eval_I(\neg)$ ]

$\forall I, eval_I(F) = 1 - eval_I(\neg F)$

ssi [ per def. de $eval_I(\neg)$ ]

$\forall I, eval_I(F) = 1 - (1 - eval_I(F))$

ssi [ per aritmètica ]

$\forall I, eval_I(F) = eval_I(F)$

ssi

*true*

# Exercici 23

23. (dificultat 3) Lema de Substitució.

Siguin $F$, $G$, $G'$ fórmules qualssevol, amb $G \equiv G'$.
Si en $F$ substituïm una aparició de una subfórmula $G$ per $G'$ obtenim una nova fórmula $F'$ amb $F \equiv F'$.

En el exemple anterior:

$$\begin{array}{ll} F \text{ és} & p \wedge (q \vee q) \\ G \text{ és} & \phantom{p \wedge {}}(q \vee q) \\ G' \text{ és} & \phantom{p \wedge (}q \\ F' \text{ és} & p \wedge q \ . \end{array}$$

# Definició de Lògica Proposicional

**Per al proper dia declasse:**

- ☞ Capítol 2 dels apunts:
  exercicis 26, 27, 28, 31, 32, 33, 34, 35, 36, 37.
- ☞ Capítol 3 dels apunts: p3.pdf