

PRÁCTICA 4 - LÓGICA PROPOSICIONAL -

Ejercicio 1. Sea $v : \mathbf{Prop} \rightarrow \{0, 1\}$ una valuación, donde \mathbf{Prop} denota el conjunto de símbolos proposicionales del cálculo proposicional. Si sólo se conocen $v(p_1), v(p_2)$ y $v(p_3)$, siendo $v(p_1) = v(p_2) = v(p_3) = 0$, argumentar si es posible decidir $v \models \alpha$ o $v \not\models \alpha$ en los siguientes casos:

- $\alpha = \neg p_1$.
- $\alpha = ((p_5 \vee p_3) \rightarrow p_1)$.
- $\alpha = ((p_1 \vee p_2) \rightarrow p_3)$.
- $\alpha = \neg p_4$.
- $\alpha = ((p_8 \rightarrow p_5) \rightarrow (p_8 \wedge p_0))$.

Ejercicio 2. Dadas las siguientes fórmulas del cálculo proposicional:

- $\alpha_1 = (\neg p_1 \rightarrow (p_3 \vee p_4))$.
- $\alpha_2 = \neg(p_2 \rightarrow (p_3 \wedge p_1))$.
- $\alpha_3 = ((\neg p_2 \rightarrow p_3) \rightarrow (p_2 \vee (p_5 \rightarrow p_3)))$.

Hallar todas las valuaciones v tales que:

- $v \models \alpha_i$.
- $v \models \alpha_i$ y $v(p_j) = 0$ si $p_j \notin \mathbf{Var}(\alpha)$.

donde \mathbf{Var} denota al conjunto de variables proposicionales y $\mathbf{Var}(\alpha)$ al subconjunto de \mathbf{Var} cuyos elementos son las variables proposicionales que aparecen en α .

Ejercicio 3. Sean $\alpha, \beta \in \mathbf{Form}$. Decimos que α es satisfacible cuando existe una valuación v tal que $v \models \alpha$. Demostrar que:

- α es tautología si y solo si $\neg \alpha$ no es satisfacible.
- $(\alpha \wedge \beta)$ es tautología si y sólo si α y β son tautologías.
- $(\alpha \vee \beta)$ es contradicción si y sólo si α y β son contradicciones.
- $(\alpha \rightarrow \beta)$ es contradicción si y sólo si α es tautología y β es contradicción.

Ejercicio 4. Sean $\alpha, \beta \in \mathbf{Form}$.

- Probar que si $\alpha \wedge \beta$ es una contingencia, entonces α es contingencia o β es contingencia.
- Dadas dos valuaciones v, v' , probar que si $v(p_i) = v'(p_i)$ para toda $p_i \in \mathbf{Var}(\alpha)$ entonces $v \models \alpha$ si y sólo si $v' \models \alpha$.
- Usando el resultado anterior, mostrar que si $\mathbf{Var}(\alpha) \cap \mathbf{Var}(\beta) = \emptyset$, entonces $(\alpha \rightarrow \beta)$ es tautología si y sólo si α es contradicción ó β es tautología.
- Análogamente, probar que si α y β son contingencias y no tienen variables proposicionales en común, entonces $\alpha \wedge \beta$ es contingencia.

Ejercicio 5. Se dice que un conjunto de conectivos es *adecuado* si con ellos se puede representar cualquier función booleana.

- Demostrar que $\{\neg, \wedge, \vee\}$ es un conjunto adecuado de operadores (sin suponer que otro conjunto es adecuado).
- Probar, usando el resultado anterior, que también son adecuados $\{\neg, \wedge\}$, $\{\neg, \vee\}$ y $\{\neg, \rightarrow\}$.
- Demostrar que $\{\neg\}$, $\{\vee, \wedge\}$ y $\{\vee, \rightarrow\}$ no son adecuados.

Ejercicio 6. Dadas $\alpha, \beta \in \mathbf{Form}$ puede escribirse $(\neg\alpha \vee \neg\beta)$ como $\alpha|\beta$ (llamada barra de *Sheffer*), y $(\neg\alpha \wedge \neg\beta)$ como $\alpha\downarrow\beta$ (barra de *Nicod*).

- Construir las tablas de verdad de $\alpha|\beta$ y $\alpha\downarrow\beta$.
- Mostrar que $\{|\}$ y $\{\downarrow\}$ son adecuados.
- Probar que si $*$ es un conector binario adecuado, entonces $*$ es $|$ ó \downarrow .

Ejercicio 7. Sean \top y \perp dos conectivos de aridad cero (i.e., constantes booleanas), que cumplen $v \models \top$ y $v \not\models \perp$ para toda valuación v .

- Probar que $\{\rightarrow, \perp\}$ es un conjunto adecuado de conectivos.
- Probar que $\{\rightarrow, \top\}$ *no* es un conjunto adecuado de conectivos.

Ejercicio 8. Sea $\mathcal{L} = \{|\}$. Podemos dar una codificación biyectiva $\# : \mathbf{Form}(\mathcal{L}) \rightarrow \mathbb{N}$ de la siguiente manera

$$\#\varphi = \begin{cases} 2 * (i - 1) & \text{si } \varphi = p_i \text{ (con } i \geq 1) \\ 2 * \langle \#\alpha, \#\beta \rangle + 1 & \text{si } \varphi = \alpha|\beta \end{cases}$$

- Probar que existe un algoritmo primitivo recursivo que resuelve el problema de *model checking proposicional*. Es decir, dada una fórmula α y una valuación *finita* v tal que $\mathbf{Var}(\alpha) \subseteq \text{Dom } v$ decide si $v \models \alpha$.
Sugerencia: Dar una codificación biyectiva para las valuaciones y justificar la elección. Recordar que existen varios esquemas de recursión que resultan ser primitivos recursivos.
- Probar que existe un algoritmo primitivo recursivo que resuelve el problema de *satisfacción proposicional*. Es decir, dada una fórmula α decide si existe una valuación v tal que $v \models \alpha$.
- Probar que existe un algoritmo primitivo recursivo que, dada una fórmula α , decide si α es una tautología.

Ejercicio 9. Sea $\mathcal{L} = \{\wedge, \vee, \neg\}$ y α una fórmula proposicional del lenguaje \mathcal{L} . Sea α^* la fórmula que resulta de reemplazar en α : $\wedge \mapsto \vee$, $\vee \mapsto \wedge$ y para todo i , $p_i \mapsto \neg p_i$. Probar que para toda valuación v , $v \models \alpha^*$ si y sólo si $v \not\models \alpha$.

Ejercicio 10. Dada una valuación v , sean p y q dos proposiciones tales que $v(p) = v(q)$. Demostrar que $v \models \varphi$ si y $v \models \varphi[p \mapsto q]$ para toda fórmula φ , donde $\varphi[p \mapsto q]$ denota la fórmula que resulta de reemplazar uniformemente la proposición p por q en φ .

Ejercicio 11. Dado un conjunto de fórmulas Γ , llamamos $\mathbf{Con}(\Gamma)$ al conjunto de consecuencias semánticas de Γ definido como $\mathbf{Con}(\Gamma) = \{\varphi : \Gamma \models \varphi\}$. Sean $\Gamma, \Gamma_1, \Gamma_2, \Gamma_3$ conjuntos de fórmulas. Probar que:

- $\Gamma \subseteq \mathbf{Con}(\Gamma)$.
- si $\Gamma_1 \subseteq \Gamma_2$, entonces $\mathbf{Con}(\Gamma_1) \subseteq \mathbf{Con}(\Gamma_2)$.
- si $\Gamma_1 \subseteq \mathbf{Con}(\Gamma_2)$ y $\Gamma_2 \subseteq \mathbf{Con}(\Gamma_3)$ entonces $\Gamma_1 \subseteq \mathbf{Con}(\Gamma_3)$.
- $\mathbf{Con}(\mathbf{Con}(\Gamma)) = \mathbf{Con}(\Gamma)$.

Ejercicio 12. Sean $\alpha, \beta \in \mathbf{Form}$.

- a. Probar que $\mathbf{Con}(\{\beta\}) \subseteq \mathbf{Con}(\{\alpha\})$ si y sólo si $\alpha \rightarrow \beta$ es tautología.
- b. Analizar la validez de las siguientes afirmaciones:
 - 1) $\mathbf{Con}(\{(\alpha \wedge \beta)\}) = \mathbf{Con}(\{\alpha\}) \cap \mathbf{Con}(\{\beta\})$.
 - 2) $\mathbf{Con}(\{(\alpha \vee \beta)\}) = \mathbf{Con}(\{\alpha\}) \cup \mathbf{Con}(\{\beta\})$.
 - 3) $\mathbf{Con}(\{(\alpha \rightarrow \beta)\}) \subseteq \mathbf{Con}(\{\beta\})$.

Ejercicio 13. * Sea $\Gamma \subseteq \mathbf{Form}$.

- a. Probar que si Γ es satisfacible y $\Gamma' \subseteq \Gamma$, entonces Γ' es satisfacible. Mostrar que la recíproca no es cierta.
- b. Probar que Γ es satisfacible si y sólo si $\mathbf{Con}(\Gamma)$ es satisfacible.
- c. ¿Es cierto que para toda fórmula α sucede $\Gamma \models \alpha$ o $\Gamma \models \neg\alpha$?

Ejercicio 14. Demostrar que son equivalentes:

- a. $\neg(\alpha_1 \wedge \dots \wedge \alpha_n) \in \mathbf{Con}(\emptyset)$.
- b. $\alpha_1, \dots, \alpha_n$ no son simultáneamente válidas para ninguna valuación.
- c. Existe una fórmula β tal que $\beta \in \mathbf{Con}(\{\alpha_1, \dots, \alpha_n\})$ y $\neg\beta \in \mathbf{Con}(\{\alpha_1, \dots, \alpha_n\})$.
- d. $\beta \in \mathbf{Con}(\{\alpha_1, \dots, \alpha_n\})$ para toda fórmula β .

*Este ejercicio puede ser entregado, de manera opcional, como se resolvería en un examen, a modo de práctica para el parcial.