

EJERCICIO 1: Sea A una fórmula.

- a) Escriba el pseudocódigo de una función recursiva que elimine las dobles negaciones de A , viendo a A como un árbol.
- b) Escriba el paso a paso para $q \wedge \neg\neg(\neg\neg p \vee r)$.
- c) Escriba el pseudocódigo de una función que elimine las dobles negaciones de A , viendo a A como una cadena de caracteres.

EJERCICIO 2: Sea A una fórmula y sea ℓ el número de ocurrencias de letras proposicionales en A , m el número de ocurrencias de conectivos binarios en A y n el número de ocurrencias de negaciones en A . Demuestre que la longitud de A , como cadena de caracteres, es $\ell + 3m + n$. [Ayuda: demuestre por inducción estructural sobre A y recurra a la definición de $\text{INORDER}(A)$.]

EJERCICIO 3: Compruebe que el resultado del algoritmo Tseitin aplicado a p es p .

EJERCICIO 4: Compruebe que el resultado del algoritmo Tseitin aplicado a $\neg p$ es $x_1 \wedge (\neg x_1 \vee \neg p) \wedge (x_1 \vee p)$.

EJERCICIO 5: Compruebe que el resultado del algoritmo Tseitin aplicado a $p \wedge q$ es $x_1 \wedge (p \vee \neg x_1) \wedge (q \vee \neg x_1) \wedge (\neg p \vee \neg q \vee x_1)$.

EJERCICIO 6: Implemente en Python el algoritmo de transformación de Tseitin.

EJERCICIO 7: Implemente en Python el algoritmo de obtención de forma clausal.