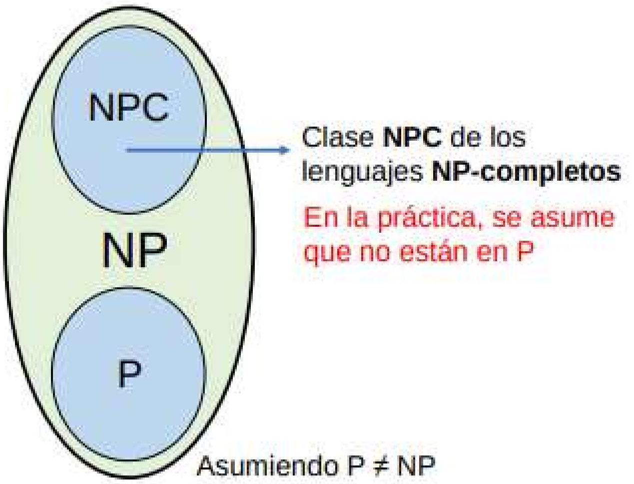
Clase 6- completitud

martes, 15 de abril de 2025 19:02

Lenguajes NP-completos.



Se rueba que si un lenguje es np completo está en p, se cumple p=np
Si se cumple p!=np, un lenguaje np completo no está en p
El lenguaje es verificable ne tiempo eficiente. Cualquier cadena se puede verificar.

Un lenguaje es np completo si li
L está en no
Y si se cualquier lenguaje de np se reduce eficientemente a L

SAT
Si está en NP se que hay una mt que verifica en tiempo eficiente el lenguaje
Cositas

Probar que un lenguaje L es np cometo.
Primero hay que probar que pertenece a np
Después una reducción polinomial de f a sat a L

SAT = {φ | φ es una fórmula booleana sin cuantificadores y es satisfactible}

$x_1 \vee (x_2 \wedge \neg x_3 \vee x_1) \wedge (x_3 \vee \neg x_1 \wedge x_3) \vee (x_5 \vee \neg x_1) \wedge \neg x_4$

CSAT = {φ | φ es una fórmula booleana sin cuantificadores en la forma normal conjuntiva (FNC) y es satisfactible}

$x_1 \wedge (x_1 \vee x_2 \vee x_5 \vee \neg x_3) \wedge (\neg x_5 \vee \neg x_3) \wedge (x_1 \vee x_2 \vee x_4 \vee \neg x_3 \vee x_3)$

conjunciones de disyunciones de literales (variables o variables negadas), conocidas como **cláusulas**

3-SAT = {φ | φ es una fórmula booleana sin cuantificadores en FNC con tres literales por cláusula y es satisfactible}

$(x_1 \vee \neg x_3 \vee x_4) \wedge (x_1 \vee x_2 \vee x_4) \wedge (\neg x_2 \vee \neg x_3 \vee x_1)$

cláusulas de tres literales