Unidad III: Complejidad Computacional

Más NP-completitud

Clase 18 - Lógica para Ciencia de la Computación/Teoría de la Computación

Prof. Miguel Romero

Ya tenemos nuestros primeros problemas NP-completos (SAT y CNF-SAT)

¿Existen más problemas NP-completos?

Los 21 problemas de Karp



Richard Karp (1935 –)

Reducibility Among Combinatorial Problems (1972).

Definición:

Una fórmula proposicional φ está en 3-CNF si φ está en CNF y cada claúsula de φ tiene a lo más 3 literales.

Definamos el siguiente lenguaje:

$$\text{3-SAT} = \{\langle\varphi\rangle \mid \varphi \text{ fórmula proposicional satisfacible en 3-CNF}\}.$$

Teorema:

3-SAT es NP-completo.

Demostración: ver ayudantía :-)

Definición:

Sea $k \geq 2$. Una fórmula proposicional φ está en k-CNF si φ está en CNF y cada claúsula de φ tiene a lo más k literales.

Para $k \geq 2$, definamos el siguiente lenguaje:

$$k\text{-SAT} = \{\langle \varphi \rangle \mid \varphi \text{ fórmula proposicional satisfacible en } k\text{-CNF}\}.$$

Corolario:

k-SAT es NP-completo, para todo $k \geq 4$.

¿Qué pasa con 2-SAT?

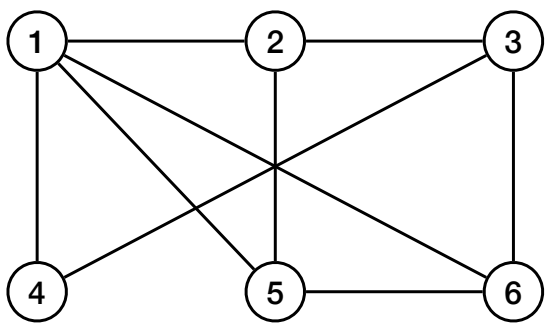
Definición:

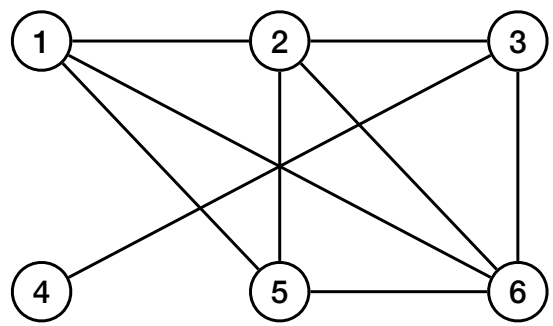
Sea $G = (V, E)$ un grafo (no dirigido). Un **clique** es un subconjunto de nodos $D \subseteq V$ tal que:

para todo par de nodos $u \neq v$ en D, se tiene que $\{u, v\} \in E$.

Idea: los nodos de un clique D están conectados todos con todos.

Ejemplo: ¿Tienen los siguientes grafos cliques de 3, 4 o 5 nodos?





CLIQUE es NP-completo

Definamos el siguiente lenguaje:

$$\text{CLIQUE} = \{\langle G, k\rangle \mid G \text{ grafo no dirigido que tiene un clique de } k \text{ nodos}\}.$$

Teorema:

CLIQUE es NP-completo.

CLIQUE está en NP (¿por qué?). Basta ver que CLIQUE es NP-hard.

CNF-SAT $\leq_p$ CLIQUE: idea

Veamos que CNF-SAT $\leq_p$ CLIQUE.

La reducción debe mapear un φ en CNF-SAT a un par $\langle G_\varphi, k_\varphi \rangle$ tal que:

$$\langle \varphi \rangle \in \text{CNF-SAT} \iff \langle G_\varphi, k_\varphi \rangle \in \text{CLIQUE}$$

Idea:

Sea $\varphi = C_1 \wedge \cdots \wedge C_m$ una fórmula proposicional en CNF-SAT.

φ es satisfacible

$\iff$

podemos escoger literales $\ell_1, \ldots, \ell_m$, uno en cada cláusula,
tal que ℓ_i y ℓ_j **no** son contradictorios, para todo $i \neq j$.

(literales contradictorios = uno es la negación del otro.)

¿Dónde está el clique acá?

CNF-SAT $\leq_p$ CLIQUE: construcción

Sea $\varphi = C_1 \wedge \cdots \wedge C_m$ en CNF.

La reducción computa $\langle G_\varphi, m \rangle$ a partir de φ donde G_φ se define como:

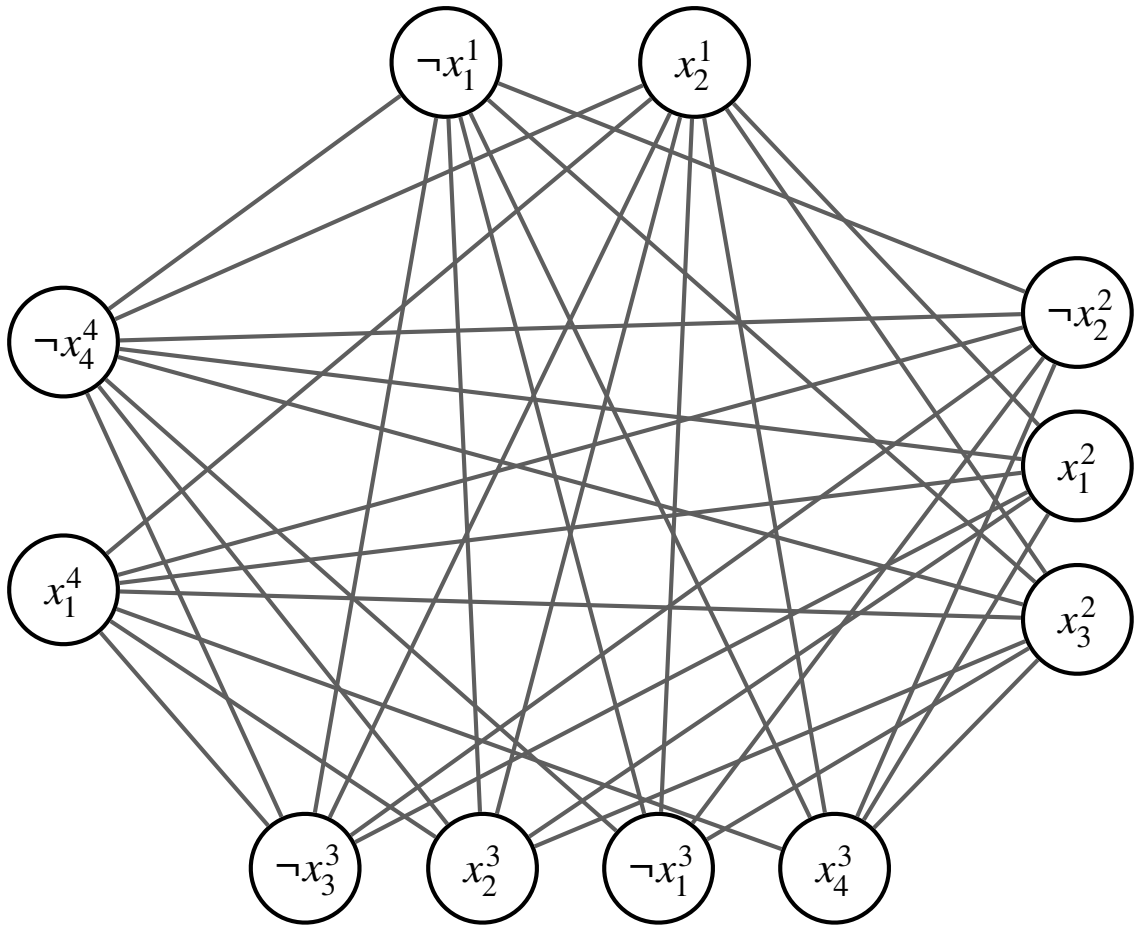
- Por cada cláusula C_i, y cada literal ℓ en C_i, hay un nodo ℓ^i en G_φ.
- $\{\ell^i, g^j\}$ es arista $\iff$ $i \neq j$ y ℓ y g son literales **no** son contradictorios.

Notar que la reducción se puede computar en tiempo polinomial.

CNF-SAT $\leq_p$ CLIQUE: ejemplo

Sea $\varphi = (\neg x_1 \vee x_2) \wedge (\neg x_2 \vee x_1 \vee x_3) \wedge (\neg x_3 \vee x_2 \vee \neg x_1 \vee x_4) \wedge (\neg x_4 \vee x_1)$.

La reducción computa $\langle G_\varphi, 4 \rangle$ donde G_φ es:



CNF-SAT $\leq_p$ CLIQUE: correctitud

$$\langle \varphi \rangle \in \text{CNF-SAT} \iff \langle G_\varphi, m \rangle \in \text{CLIQUE}.$$

($\Rightarrow$)

Si $\langle \varphi \rangle \in$ CNF-SAT, existe una valuación σ que satisface a $\varphi = C_1 \wedge \cdots \wedge C_m$.

Para cada cláusula C_i, existe un literal ℓ_i tal que $\sigma(\ell_i) = 1$.

Sigue que ℓ_i y ℓ_j son literales no contradictorios, para todo $i \neq j$.

Por construcción de G_φ,
el subconjunto de nodos $\{\ell_i^i \mid 1 \leq i \leq m\}$ es un clique de m nodos.

Concluimos que $\langle G_\varphi, m \rangle \in$ CLIQUE.

CNF-SAT $\leq_p$ CLIQUE: correctitud

$$\langle \varphi \rangle \in \text{CNF-SAT} \iff \langle G_\varphi, m \rangle \in \text{CLIQUE}.$$

($\Leftarrow$)

Si $\langle G_\varphi, m \rangle \in$ CLIQUE, existe un clique $\{v_1, \ldots, v_m\}$ de m nodos en G_φ.

Como en G_φ no hay aristas entre nodos asociados a una misma cláusula, sigue que:

$$\{v_1, \ldots, v_m\} = \{\ell_1^1, \ldots, \ell_m^m\}.$$

donde ℓ_i es un literal en la cláusula C_i.

Por construcción de G_φ,
los literales ℓ_i y ℓ_j son literales no contradictorios, para todo $i \neq j$.

Podemos definir una valuación σ tal que $\sigma(\ell_i) = 1$, para todo $1 \leq i \leq m$.

(le asignamos cualquier valor al resto de las variables, si es que hay.)

Obtenemos que σ satisface a φ, luego $\langle \varphi \rangle \in$ CNF-SAT.

CLIQUE: comentarios

¿Qué sucede con los siguientes problemas?

- $\text{CLIQUE}_{\geq} = \{\langle G, k\rangle \mid G \text{ grafo que tiene un clique con } k \text{ o más nodos}\}$.
- $\text{CLIQUE}_{\leq} = \{\langle G, k\rangle \mid G \text{ grafo que tiene un clique con } k \text{ o menos nodos}\}$.
- Para $k \geq 1$:

$$\text{CLIQUE}_k = \{\langle G\rangle \mid G \text{ grafo que tiene un clique con } k \text{ nodos}\}.$$

Definición:

Sea $G = (V, E)$ un grafo (no dirigido). Un **independiente** es un subconjunto de nodos $I \subseteq V$ tal que:

para todo par de nodos $u \neq v$ en I, se tiene que $\{u, v\} \notin E$.

Idea: los nodos de un independiente I **no** están conectados par a par.

INDEPENDENT-SET es NP-completo

Definamos el siguiente lenguaje:

INDEPENDENT-SET = $\{\langle G, k\rangle \mid G$ grafo que tiene un independiente de k nodos$\}$.

Teorema:
INDEPENDENT-SET es NP-completo.

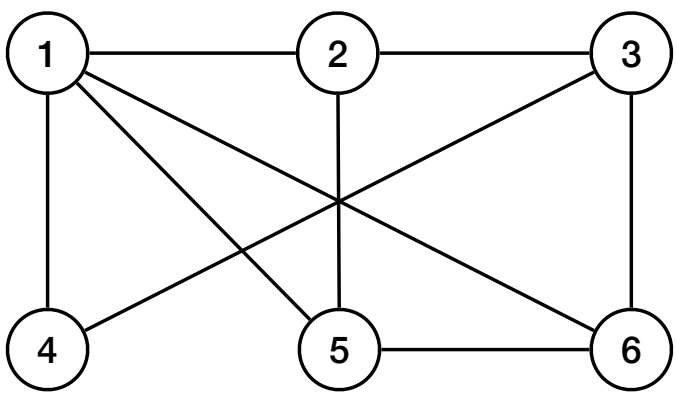
INDEPENDENT-SET está en NP (¿por qué?). Basta ver que INDEPENDENT-SET es NP-hard.

Definición:

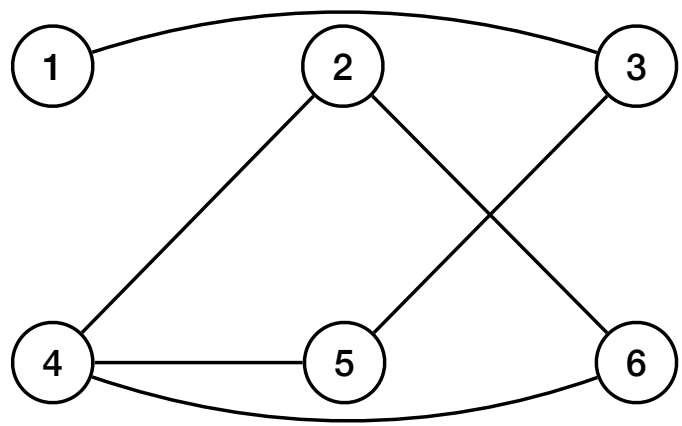
Sea $G = (V, E)$ un grafo. El grafo complemento $\overline{G}$ de G se define como $\overline{G} = (V, \overline{E})$ donde para todo $u, v \in V$ se cumple:

$$\{u, v\} \in \overline{E} \iff \{u, v\} \notin E.$$

Ejemplo:



G



$\overline{G}$

Definición:

Sea $G = (V, E)$ un grafo. El grafo complemento $\overline{G}$ de G se define como $\overline{G} = (V, \overline{E})$ donde para todo $u, v \in V$ se cumple:

$$\{u, v\} \in \overline{E} \iff \{u, v\} \notin E.$$

Proposición:

Sea $G = (V, E)$ un grafo y $D \subseteq V$ un subconjunto de nodos. Se cumple que:

$$D \text{ es un clique en } G \iff D \text{ es un independiente en } \overline{G}.$$

CLIQUE $\leq_p$ INDEPENDENT-SET

Veamos que CLIQUE $\leq_p$ INDEPENDENT-SET.

La reducción mapea un par $\langle G, k \rangle$ al par $\langle \overline{G}, k \rangle$.

Notar que la reducción se puede computar en tiempo polinomial.

Se tiene que:

$$\langle G, k \rangle \in \text{CLIQUE} \iff \langle \overline{G}, k \rangle \in \text{INDEPENDENT-SET}.$$

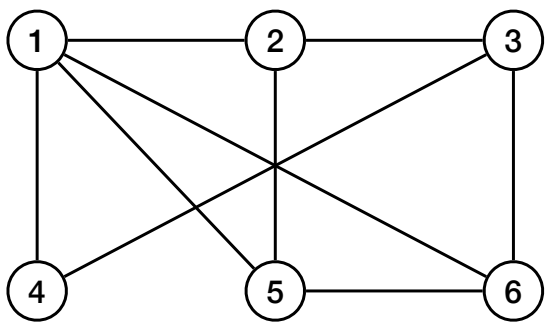
Definición:

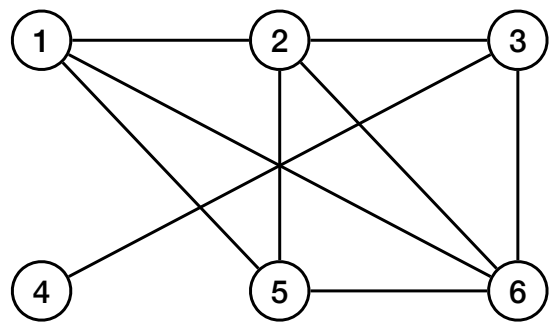
Sea $G = (V, E)$ un grafo (no dirigido). Un **vertex cover** es un subconjunto de nodos $D \subseteq V$ tal que:

$$\text{para toda arista } \{u, v\} \in E, \text{ se tiene que } u \in D \text{ o } v \in D.$$

Idea: los nodos de un vertex cover D cubren todas las aristas.

Ejemplo: ¿Tienen los siguientes grafos vertex covers de 3, 4 o 5 nodos?





VERTEX-COVER es NP-completo

Definamos el siguiente lenguaje:

$$\text{VERTEX-COVER} = \{\langle G, k\rangle \mid G \text{ grafo que tiene vertex cover de } k \text{ nodos}\}.$$

Teorema:
VERTEX-COVER es NP-completo.

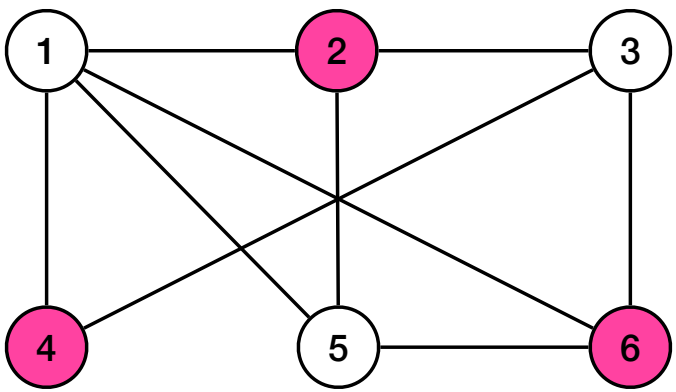
VERTEX-COVER está en NP (¿por qué?).
Basta ver que VERTEX-COVER es NP-hard.

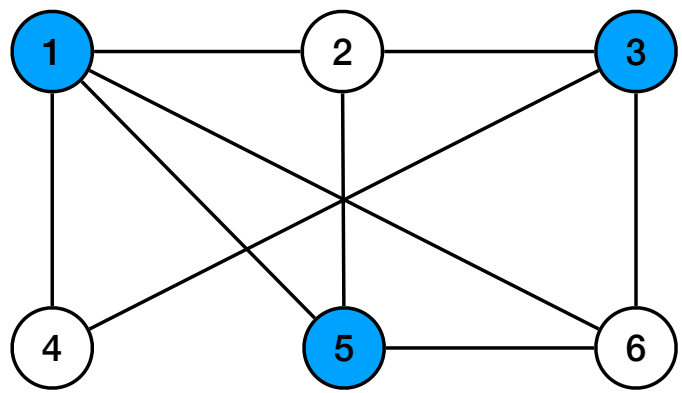
Proposición:

Sea $G = (V, E)$ un grafo e $I \subseteq V$ un subconjunto de nodos. Se cumple que:

$$I \text{ es un independiente en } G \iff V \setminus I \text{ es un vertex cover en } G.$$

Ejemplo:





Proposición:

Sea $G = (V, E)$ un grafo e $I \subseteq V$ un subconjunto de nodos. Se cumple que:

$$I \text{ es un independiente en } G \iff V \setminus I \text{ es un vertex cover en } G.$$

Demostración:

$$I \text{ es un independiente en } G = (V, E)$$
$$\iff$$
$$\{u, v\} \notin E, \text{ para todo } u \neq v \text{ en } I$$
$$\iff$$
$$\text{para toda arista } \{u, v\} \in E, \text{ se cumple } u \in V \setminus I \text{ o } v \in V \setminus I$$
$$\iff$$
$$V \setminus I \text{ es un vertex cover en } G.$$

INDEPENDENT-SET $\leq_p$ VERTEX-COVER

Veamos que INDEPENDENT-SET $\leq_p$ VERTEX-COVER.

La reducción mapea un par $\langle G, k \rangle$ al par $\langle G, n - k \rangle$,
donde n es la cantidad de nodos de G.

Notar que la reducción se puede computar en tiempo polinomial.

Se tiene que:

$$\langle G, k \rangle \in \text{INDEPENDENT-SET} \iff \langle G, n - k \rangle \in \text{VERTEX-COVER}.$$

VERTEX-COVER: comentarios

¿Qué sucede con los siguientes problemas?

- VERTEX-COVER$_{\geq}$ = $\{\langle G, k\rangle \mid G$ grafo con un vertex cover de k o más nodos$\}$.
- VERTEX-COVER$_{\leq}$ = $\{\langle G, k\rangle \mid G$ grafo con un vertex cover de k o menos nodos$\}$.
- Para $k \geq 1$:

  VERTEX-COVER$_k$ = $\{\langle G\rangle \mid G$ grafo con un vertex cover de k nodos$\}$.

Definición:

Sea $\mathcal{S}$ una colección de conjuntos finitos. Un **set cover** para $\mathcal{S}$ es una subcolección $\mathcal{C} \subseteq \mathcal{S}$ tal que:

$$\bigcup_{S \in \mathcal{S}} S = \bigcup_{C \in \mathcal{C}} C.$$

Idea: el set cover $\mathcal{C}$ cubre los mismos elementos que S.

Ejemplo:

Sea $\mathcal{S} = \{\{1,2,3\},\{2,4,5\},\{1,4\},\{3,5\},\{1,3,4\}\}$.

- $\mathcal{C} = \{\{1,2,3\},\{1,4\},\{3,5\}\}$ es set cover (de tamaño 3), ya que:

$$\bigcup_{C \in \mathcal{C}} C = \{1,2,3,4,5\} = \bigcup_{S \in \mathcal{S}} S.$$

- $\mathcal{C} = \{\{2,4,5\},\{1,3,4\}\}$ también es set cover (de tamaño 2).
- $\mathcal{C} = \{\{1,2,3\},\{1,4\},\{1,3,4\}\}$ no es set cover ya que:

$$\bigcup_{C \in \mathcal{C}} C = \{1,2,3,4\} \neq \{1,2,3,4,5\} = \bigcup_{S \in \mathcal{S}} S.$$

Definición:

Sea $\mathcal{S}$ una colección de conjuntos finitos. Un **set cover** para $\mathcal{S}$ es una subcolección $\mathcal{C} \subseteq \mathcal{S}$ tal que:

$$\bigcup_{S \in \mathcal{S}} S = \bigcup_{C \in \mathcal{C}} C.$$

Idea: el set cover $\mathcal{C}$ cubre los mismos elementos que S.

Observación:

Sea $\mathcal{S}$ una colección de conjuntos finitos tal que $U = \bigcup_{S \in \mathcal{S}} S$. (el universo.)

$$\mathcal{C} \subseteq \mathcal{S} \text{ es un set cover } \iff \text{ para todo } u \in U \text{ se tiene } u \in \bigcup_{C \in \mathcal{C}} C.$$

SET-COVER es NP-completo

Definamos el siguiente lenguaje:

SET-COVER =

$\{\langle \mathcal{S}, k \rangle \mid \mathcal{S}$ colección de conjuntos finitos con un set cover de tamaño $k\}$.

Teorema:

SET-COVER es NP-completo.

SET-COVER está en NP (¿por qué?).

Basta ver que SET-COVER es NP-hard.

VERTEX-COVER $\leq_p$ SET-COVER

Veamos que VERTEX-COVER $\leq_p$ SET-COVER.

La reducción mapea un par $\langle G, k \rangle$, donde $G = (V, E)$, a un par $\langle \mathcal{S}_G, k \rangle$, donde la colección $\mathcal{S}$ se define como:

- Para cada nodo $v \in V$, definimos el conjunto S_v como:

$$S_v = \{e \in E \mid e \text{ es incidente a } v\}.$$

- Si $V = \{v_1, \ldots, v_n\}$, definimos $\mathcal{S}_G = \{S_{v_1}, \ldots, S_{v_n}\}$.

La reducción se puede computar en tiempo polinomial.

VERTEX-COVER $\leq_p$ SET-COVER: correctitud

$$\langle G, k \rangle \in \text{VERTEX-COVER} \iff \langle \mathcal{S}_G, k \rangle \in \text{SET-COVER}.$$

($\Rightarrow$)

Si $\langle G, k \rangle \in$ VERTEX-COVER, donde $G = (V, E)$,
existe vertex cover $D = \{z_1, \ldots, z_k\} \subseteq V$ con k nodos.

Veamos que $\mathcal{C} = \{S_{z_1}, \ldots, S_{z_k}\}$ es un set cover para $\mathcal{S}_G = \{S_{v_1}, \ldots, S_{v_n}\}$.

(basta ver que para toda arista $e \in E$ se cumple que $e \in \bigcup_{j=1}^{k} S_{z_j}$.)

- Sea $e \in E$ arbitraria. Como $D = \{z_1, \ldots, z_k\}$ es vertex cover, existe $j \in \{1, \ldots, k\}$ tal que la arista e es incidente $z_j \in D$.
- Por definición de S_{z_j}, sigue que $e \in S_{z_j}$, y luego $e \in \bigcup_{j=1}^{k} S_{z_j}$.

Concluimos que $\mathcal{S}_G$ tiene un set cover de tamaño k.

VERTEX-COVER $\leq_p$ SET-COVER: correctitud

$$\langle G, k \rangle \in \text{VERTEX-COVER} \iff \langle \mathcal{S}_G, k \rangle \in \text{SET-COVER}.$$

($\Leftarrow$)

Si $\langle \mathcal{S}_G, k \rangle \in$ SET-COVER, existe set cover $\mathcal{C} \subseteq \mathcal{S}_G = \{S_{v_1}, \ldots, S_{v_n}\}$ de tamaño k.
($G = (V, E)$ y $V = \{v_1, \ldots, v_n\}$.)

Tenemos $\mathcal{C} = \{S_{z_1}, \ldots, S_{z_k}\}$ para cierto subconjunto de nodos $D = \{z_1, \ldots, z_k\} \subseteq V$.

Veamos que $D = \{z_1, \ldots, z_k\}$ es un vertex cover para G.

- Sea $e \in E$ arbitraria. Como $\mathcal{C} = \{S_{z_1}, \ldots, S_{z_k}\}$ es set cover, $e \in \bigcup_{j=1}^{k} S_{z_j}$, y luego existe $j \in \{1, \ldots, k\}$ tal que $e \in S_{z_j}$.
- Por definición de S_{z_j}, tenemos que e es incidente a z_j.
- Luego toda arista es incidente a algún nodo de $D = \{z_1, \ldots, z_k\}$.

Concluimos que G tiene un vertex cover con k nodos.

Definamos el siguiente lenguaje:

SUBSET-SUM =

$\{\langle A, t\rangle \mid A$ multi-conjunto de números naturales tal que existe $S \subseteq A$ con $\sum_{s \in S} s = t\}$.

(decimos que S es una solución para A y t.)

Ejemplos:

¿Cuáles de las siguientes instancias están en SUBSET-SUM?

$\langle\{3, 2, 2, 5, 10, 10\}, 19\rangle$ $\langle\{3, 2, 2, 5, 10, 10\}, 21\rangle$

Definamos el siguiente lenguaje:

SUBSET-SUM =

$\{\langle A, t\rangle \mid A$ multi-conjunto de números naturales tal que existe $S \subseteq A$ con $\sum_{s \in S} s = t\}$.

(decimos que S es una solución para A y t.)

Teorema:

SUBSET-SUM es NP-completo.

SUBSET-SUM está en NP (¿por qué?).
Basta ver que SUBSET-SUM es NP-hard.

3-SAT $\leq_p$ SUBSET-SUM

Veamos que 3-SAT $\leq_p$ SUBSET-SUM.

Sea $\varphi = C_1 \wedge \cdots \wedge C_m$ una fórmula en 3-CNF sobre variables $\{x_1, \ldots x_n\}$.

La reducción computa $\langle A_\varphi, t_\varphi \rangle$ donde A_φ se define como:

- Para cada $1 \leq i \leq n$, tenemos dos números v_i y $\overline{v}_i$ en A_φ.
  (v_i y $\overline{v}_i$ están asociados a x_i y $\neg x_i$, respectivamente.)
- Para cada $1 \leq j \leq m$, tenemos dos números a_j y b_j en A_φ.
- Visualizaremos todos estos números como números de $n + m$ dígitos.

3-SAT $\leq_p$ SUBSET-SUM

		x_1	x_2	x_3	x_4	$\cdots$	x_n	C_1	C_2	$\cdots$	C_m
x_1	v_1	1	0	0	0	$\cdots$	0	1	0	$\cdots$	0
$\neg x_1$	$\overline{v}_1$	1	0	0	0	$\cdots$	0	0	0	$\cdots$	1
x_2	v_2		1	0	0	$\cdots$	0	0	1	$\cdots$	0
$\neg x_2$	$\overline{v}_2$		1	0	0	$\cdots$	0	1	0	$\cdots$	0
x_3	v_3			1	0	$\cdots$	0	1	1	$\cdots$	0
$\neg x_3$	$\overline{v}_3$			1	0	$\cdots$	0	0	0	$\cdots$	0
$\vdots$	$\vdots$					$\ddots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	v_n						1	0	0	$\cdots$	0
$\neg x_n$	$\overline{v}_n$						1	0	0	$\cdots$	0
	a_1							1	0	$\cdots$	0
	b_1							1	0	$\cdots$	0
	a_2								1	$\cdots$	0
	b_2								1	$\cdots$	0
	$\vdots$									$\ddots$	$\vdots$
	a_m										1
	b_m										1
	t_φ	1	1	1	1	$\cdots$	1	3	3	$\cdots$	3

Los números de A_φ se definen según la tabla:

cada fila corresponde a un número, cada columna a un dígito.

3-SAT $\leq_p$ SUBSET-SUM

		x_1	x_2	x_3	x_4	$\cdots$	x_n	C_1	C_2	$\cdots$	C_m
x_1	v_1	1	0	0	0	$\cdots$	0	1	0	$\cdots$	0
$\neg x_1$	$\overline{v}_1$	1	0	0	0	$\cdots$	0	0	0	$\cdots$	1
x_2	v_2		1	0	0	$\cdots$	0	0	1	$\cdots$	0
$\neg x_2$	$\overline{v}_2$		1	0	0	$\cdots$	0	1	0	$\cdots$	0
x_3	v_3			1	0	$\cdots$	0	1	1	$\cdots$	0
$\neg x_3$	$\overline{v}_3$			1	0	$\cdots$	0	0	0	$\cdots$	0
$\vdots$	$\vdots$					$\ddots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	v_n						1	0	0	$\cdots$	0
$\neg x_n$	$\overline{v}_n$						1	0	0	$\cdots$	0
	a_1							1	0	$\cdots$	0
	b_1							1	0	$\cdots$	0
	a_2								1	$\cdots$	0
	b_2								1	$\cdots$	0
	$\vdots$									$\ddots$	$\vdots$
	a_m										1
	b_m										1
	t_φ	1	1	1	1	$\cdots$	1	3	3	$\cdots$	3

Para cada $1 \leq j \leq m$, la columna j del cuadrante arriba-derecha se define como:

un dígito es 1 $\iff$ el literal correspondiente aparece en la cláusula C_j.

Ejemplo: $C_1 = (x_1 \vee \neg x_2 \vee x_3)$ $C_2 = (x_2 \vee x_3)$ $C_m = (\neg x_1)$

3-SAT $\leq_p$ SUBSET-SUM

		x_1	x_2	x_3	x_4	$\cdots$	x_n	C_1	C_2	$\cdots$	C_m
x_1	v_1	1	0	0	0	$\cdots$	0	1	0	$\cdots$	0
$\neg x_1$	$\overline{v}_1$	1	0	0	0	$\cdots$	0	0	0	$\cdots$	1
x_2	v_2		1	0	0	$\cdots$	0	0	1	$\cdots$	0
$\neg x_2$	$\overline{v}_2$		1	0	0	$\cdots$	0	1	0	$\cdots$	0
x_3	v_3			1	0	$\cdots$	0	1	1	$\cdots$	0
$\neg x_3$	$\overline{v}_3$			1	0	$\cdots$	0	0	0	$\cdots$	0
$\vdots$	$\vdots$					$\ddots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	v_n						1	0	0	$\cdots$	0
$\neg x_n$	$\overline{v}_n$						1	0	0	$\cdots$	0
	a_1							1	0	$\cdots$	0
	b_1							1	0	$\cdots$	0
	a_2								1	$\cdots$	0
	b_2								1	$\cdots$	0
	$\vdots$									$\ddots$	$\vdots$
	a_m										1
	b_m										1
	t_φ	1	1	1	1	$\cdots$	1	3	3	$\cdots$	3

En cada columna hay a lo más 5 dígitos 1,

luego para cualquier $S \subseteq A_\varphi$ al sumar los números en S nunca hay "carry".

($\Longrightarrow$ podemos analizar las columnas por separado.)

3-SAT $\leq_p$ SUBSET-SUM

		x_1	x_2	x_3	x_4	$\cdots$	x_n	C_1	C_2	$\cdots$	C_m
x_1	v_1	1	0	0	0	$\cdots$	0	1	0	$\cdots$	0
$\neg x_1$	$\overline{v}_1$	1	0	0	0	$\cdots$	0	0	0	$\cdots$	1
x_2	v_2		1	0	0	$\cdots$	0	0	1	$\cdots$	0
$\neg x_2$	$\overline{v}_2$		1	0	0	$\cdots$	0	1	0	$\cdots$	0
x_3	v_3			1	0	$\cdots$	0	1	1	$\cdots$	0
$\neg x_3$	$\overline{v}_3$			1	0	$\cdots$	0	0	0	$\cdots$	0
$\vdots$	$\vdots$					$\ddots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	v_n						1	0	0	$\cdots$	0
$\neg x_n$	$\overline{v}_n$						1	0	0	$\cdots$	0
	a_1							1	0	$\cdots$	0
	b_1							1	0	$\cdots$	0
	a_2								1	$\cdots$	0
	b_2								1	$\cdots$	0
	$\vdots$									$\ddots$	$\vdots$
	a_m										1
	b_m										1
	t_φ	1	1	1	1	$\cdots$	1	3	3	$\cdots$	3

En toda solución S para A_φ y t_φ, para cada $1 \leq i \leq n$, tenemos que:

$v_i \in S$ o $\overline{v}_i \in S$, pero no ambas!

S representa una valuación σ_S: $\sigma_S(\ell) = 1 \iff$ el literal ℓ participa en S.

3-SAT $\leq_p$ SUBSET-SUM

		x_1	x_2	x_3	x_4	$\cdots$	x_n	C_1	C_2	$\cdots$	C_m
x_1	v_1	1	0	0	0	$\cdots$	0	1	0	$\cdots$	0
$\neg x_1$	$\overline{v}_1$	1	0	0	0	$\cdots$	0	0	0	$\cdots$	1
x_2	v_2		1	0	0	$\cdots$	0	0	1	$\cdots$	0
$\neg x_2$	$\overline{v}_2$		1	0	0	$\cdots$	0	1	0	$\cdots$	0
x_3	v_3			1	0	$\cdots$	0	1	1	$\cdots$	0
$\neg x_3$	$\overline{v}_3$			1	0	$\cdots$	0	0	0	$\cdots$	0
$\vdots$	$\vdots$					$\ddots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	v_n						1	0	0	$\cdots$	0
$\neg x_n$	$\overline{v}_n$						1	0	0	$\cdots$	0
	a_1							1	0	$\cdots$	0
	b_1							1	0	$\cdots$	0
	a_2								1	$\cdots$	0
	b_2								1	$\cdots$	0
	$\vdots$									$\ddots$	$\vdots$
	a_m										1
	b_m										1
	t_φ	1	1	1	1	$\cdots$	1	3	3	$\cdots$	3

En toda solución S para A_φ y t_φ, para cada $1 \leq j \leq m$, tenemos que:

la suma de los dígitos en la columna j del cuadrante arriba-derecha debe ser al menos 1.

3-SAT $\leq_p$ SUBSET-SUM

		x_1	x_2	x_3	x_4	$\cdots$	x_n	C_1	C_2	$\cdots$	C_m
x_1	v_1	1	0	0	0	$\cdots$	0	1	0	$\cdots$	0
$\neg x_1$	$\overline{v}_1$	1	0	0	0	$\cdots$	0	0	0	$\cdots$	1
x_2	v_2		1	0	0	$\cdots$	0	0	1	$\cdots$	0
$\neg x_2$	$\overline{v}_2$		1	0	0	$\cdots$	0	1	0	$\cdots$	0
x_3	v_3			1	0	$\cdots$	0	1	1	$\cdots$	0
$\neg x_3$	$\overline{v}_3$			1	0	$\cdots$	0	0	0	$\cdots$	0
$\vdots$	$\vdots$					$\ddots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	v_n						1	0	0	$\cdots$	0
$\neg x_n$	$\overline{v}_n$						1	0	0	$\cdots$	0
	a_1							1	0	$\cdots$	0
	b_1							1	0	$\cdots$	0
	a_2								1	$\cdots$	0
	b_2								1	$\cdots$	0
	$\vdots$									$\ddots$	$\vdots$
	a_m										1
	b_m										1
	t_φ	1	1	1	1	$\cdots$	1	3	3	$\cdots$	3

En toda solución S para A_φ y t_φ, para cada $1 \leq j \leq m$, tenemos que:

la suma de los dígitos en la columna j del cuadrante arriba-derecha debe ser al menos 1.

$\Longrightarrow$ cada cláusula C_j tiene un literal ℓ_j que participa en la solución S.

3-SAT $\leq_p$ SUBSET-SUM

		x_1	x_2	x_3	x_4	$\cdots$	x_n	C_1	C_2	$\cdots$	C_m
x_1	v_1	1	0	0	0	$\cdots$	0	1	0	$\cdots$	0
$\neg x_1$	$\overline{v}_1$	1	0	0	0	$\cdots$	0	0	0	$\cdots$	1
x_2	v_2		1	0	0	$\cdots$	0	0	1	$\cdots$	0
$\neg x_2$	$\overline{v}_2$		1	0	0	$\cdots$	0	1	0	$\cdots$	0
x_3	v_3			1	0	$\cdots$	0	1	1	$\cdots$	0
$\neg x_3$	$\overline{v}_3$			1	0	$\cdots$	0	0	0	$\cdots$	0
$\vdots$	$\vdots$					$\ddots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	v_n						1	0	0	$\cdots$	0
$\neg x_n$	$\overline{v}_n$						1	0	0	$\cdots$	0
	a_1							1	0	$\cdots$	0
	b_1							1	0	$\cdots$	0
	a_2								1	$\cdots$	0
	b_2								1	$\cdots$	0
	$\vdots$									$\ddots$	$\vdots$
	a_m										1
	b_m										1
	t_φ	1	1	1	1	$\cdots$	1	3	3	$\cdots$	3

En toda solución S para A_φ y t_φ, para cada $1 \leq j \leq m$, tenemos que:

la suma de los dígitos en la columna j del cuadrante arriba-derecha debe ser al menos 1.

$\Longrightarrow$ cada cláusula C_j tiene un literal ℓ_j tal que $\sigma_S(\ell_j) = 1$.

3-SAT $\leq_p$ SUBSET-SUM

		x_1	x_2	x_3	x_4	$\cdots$	x_n	C_1	C_2	$\cdots$	C_m
x_1	v_1	1	0	0	0	$\cdots$	0	1	0	$\cdots$	0
$\neg x_1$	$\overline{v}_1$	1	0	0	0	$\cdots$	0	0	0	$\cdots$	1
x_2	v_2		1	0	0	$\cdots$	0	0	1	$\cdots$	0
$\neg x_2$	$\overline{v}_2$		1	0	0	$\cdots$	0	1	0	$\cdots$	0
x_3	v_3			1	0	$\cdots$	0	1	1	$\cdots$	0
$\neg x_3$	$\overline{v}_3$			1	0	$\cdots$	0	0	0	$\cdots$	0
$\vdots$	$\vdots$					$\ddots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	v_n						1	0	0	$\cdots$	0
$\neg x_n$	$\overline{v}_n$						1	0	0	$\cdots$	0
	a_1							1	0	$\cdots$	0
	b_1							1	0	$\cdots$	0
	a_2								1	$\cdots$	0
	b_2								1	$\cdots$	0
	$\vdots$									$\ddots$	$\vdots$
	a_m										1
	b_m										1
	t_φ	1	1	1	1	$\cdots$	1	3	3	$\cdots$	3

En toda solución S para A_φ y t_φ, para cada $1 \leq j \leq m$, tenemos que:

la suma de los dígitos en la columna j del cuadrante arriba-derecha debe ser al menos 1.

S representa una valuación σ_S que satisface a φ!

La reducción se puede computar en tiempo polinomial.

Se tiene que:

$$\langle \varphi \rangle \in \text{3-SAT} \iff \langle A_\varphi, t_\varphi \rangle \in \text{SUBSET-SUM}.$$

($\Leftarrow$)

Discusión anterior.

La reducción se puede computar en tiempo polinomial.

Se tiene que:

$$\langle \varphi \rangle \in \text{3-SAT} \iff \langle A_\varphi, t_\varphi \rangle \in \text{SUBSET-SUM}.$$

($\Rightarrow$)

Si $\langle \varphi \rangle \in$ 3-SAT, existe valuación σ que satisface a φ.

Podemos escoger $S \subseteq A_\varphi$ de la siguiente forma:

- Para cada $1 \leq i \leq n$:
  - si $\sigma(x_i) = 1$, entonces $v_i \in S$.
  - si $\sigma(\neg x_i) = 1$, entonces $\overline{v}_i \in S$.
- Agregamos a S los elementos necesarios de $\{a_1, b_1, \ldots, a_m, b_m\}$, de manera que, para cada $1 \leq j \leq m$, la suma de la columna j sea 3.
  (siempre se puede, ya que σ satisface a φ.)

3-SAT $\leq_p$ SUBSET-SUM

		x_1	x_2	x_3	x_4	$\cdots$	x_n	C_1	C_2	$\cdots$	C_m
x_1	v_1	1	0	0	0	$\cdots$	0	1	0	$\cdots$	0
$\neg x_1$	$\overline{v}_1$	1	0	0	0	$\cdots$	0	0	0	$\cdots$	1
x_2	v_2		1	0	0	$\cdots$	0	0	1	$\cdots$	0
$\neg x_2$	$\overline{v}_2$		1	0	0	$\cdots$	0	1	0	$\cdots$	0
x_3	v_3			1	0	$\cdots$	0	1	1	$\cdots$	0
$\neg x_3$	$\overline{v}_3$			1	0	$\cdots$	0	0	0	$\cdots$	0
$\vdots$	$\vdots$					$\ddots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	v_n						1	0	0	$\cdots$	0
$\neg x_n$	$\overline{v}_n$						1	0	0	$\cdots$	0
	a_1							1	0	$\cdots$	0
	b_1							1	0	$\cdots$	0
	a_2								1	$\cdots$	0
	b_2								1	$\cdots$	0
	$\vdots$									$\ddots$	$\vdots$
	a_m										1
	b_m										1
	t_φ	1	1	1	1	$\cdots$	1	3	3	$\cdots$	3