Algoritmi e Strutture Dati

Lezione 33

10 dicembre 2025

PROBLEMI DI DECISIONE: alcune classi

$$P = \bigcup_{c=0}^{\infty} TIME(n^c)$$ tempo polinomiale

$$PSPACE = \bigcup_{c=0}^{\infty} SPACE(n^c)$$ spazio polinomiale

$$EXPTIME = \bigcup_{c=0}^{\infty} TIME(2^{n^c})$$ tempo esponenziale

$$P \subseteq PSPACE \subseteq EXPTIME \qquad P \neq EXPTIME$$

almeno una di queste due inclusioni è propria

Problema SODDISFACIBILITA' (SODD)

<u>Istanza</u> Formula booleana ϕ in forma normale congiuntiva con insieme di variabili V

<u>Questione</u> $\exists$ un assegnamento alle variabili in V che rende vera ϕ, cioè t.c. $\phi(f)=1$?

$$\phi(x_1, x_2, x_3) = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x}_1 \vee \overline{x}_3)$$

ALGORITMI NON DETERMINISTICI

- ISTRUZIONE indovina $b \in \{0,1\}$
- TRA LE POSSIBILI SCELTE L'ALGORITMO INDOVINA SEMPRE QUELLA "GIUSTA" (se c'è)

ALGORITMO sodd (Formula $\phi(x_1, \dots, x_n)$) $\rightarrow$ boolean

```
FOR i ← 1 TO n DO
    z_i ← indovina un valore in {0, 1}

r ← φ(z_1, z_2, ..., z_n)

RETURN r
```

Esempio

$$\phi(x_1, x_2, x_3) = (x_1 \vee x_2 \vee x_3) \wedge (\overline{x}_1 \vee \overline{x}_3)$$

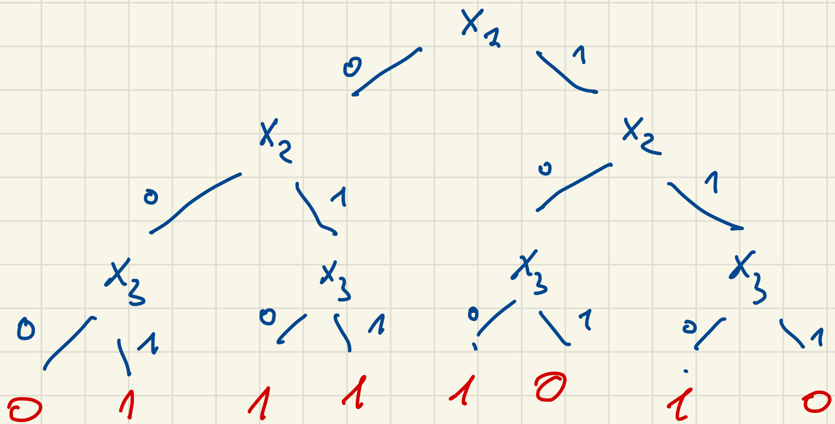
ALGORITMO sodd (Formula $\phi(x_1, \dots, x_n)$) $\longrightarrow$ boolean

indovina

```
FOR i ← 1 TO n DO
    z_i ← indovina un valore in {0, 1}

r ← φ(z_1, z_2, ..., z_n)
RETURN r
```

verifica



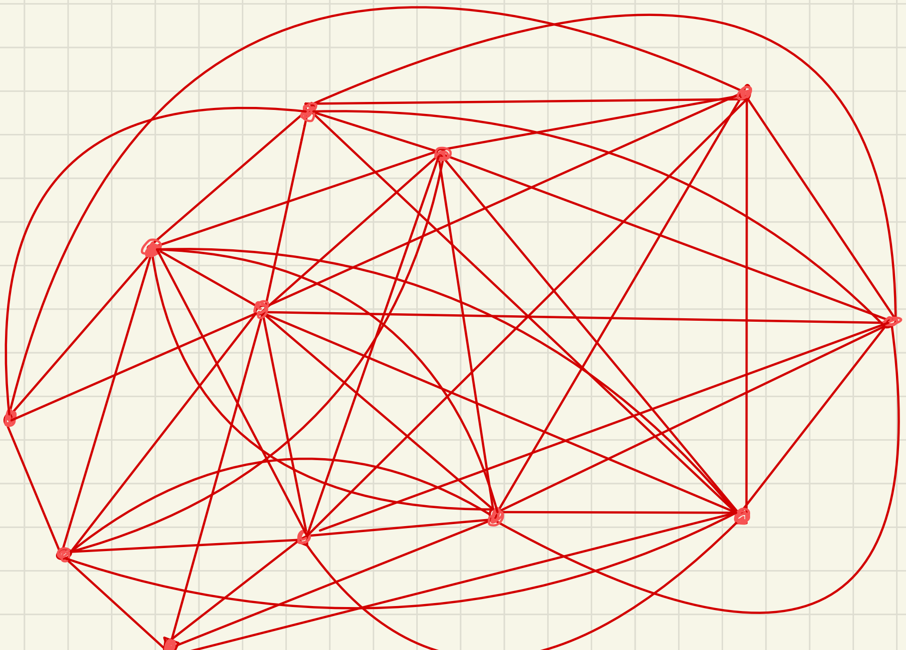
albero di computazione

Problema CLIQUE

Istanza Grafo non orientato $G = (V, E)$

Intero $K \geq 0$

Questione $\exists$ sottografo completo di G con K vertici?



Problema CLIQUE

Istanza: Grafo non orientato $G=(V,E)$

Intero $K \geq 0$

Questione: $\exists$ sottografo completo di G con K vertici?

ALGORITMO clique (Grafo $G=(V,E)$, intero K) $\rightarrow$ boolean

```
V' ← ∅
FOR EACH v ∈ V DO
    b ← indovina un valore in {0,1}
    IF b = 1 THEN V' ← V' ∪ {v}
IF #V' ≠ K THEN RETURN false
risposta ← true
FOR EACH v ∈ V' DO
    FOR EACH u ∈ V' DO
        IF u ≠ v AND (u,v) ∉ E THEN
            risposta ← false
RETURN risposta
```

indovina

verifica

CLASSE NP

NP = Classe dei problemi di decisione risolubili in tempo polinomiale da algoritmi nondeterministici

= Classe dei problemi di decisione con certificati verificabili in tempo polinomiale

NP

nondeterministico

polinomiale

SODD $\in$ NP

PROBLEMI NP

ALGORITMO

- FASE NON DETERMINISTICA — costruzione del certificato
- FASE DETERMINISTICA — verifica del certificato in tempo polinomiale

$$P \subseteq NP$$

$$P \stackrel{?}{=} NP$$

NPSPACE

Classe dei problemi di decisione risolubili in spazio polinomiale da algoritmi nondeterministici

$$\mathrm{PSPACE} = \mathrm{NPSPACE}$$

Teo di Savitch [1970]

$$P \subseteq NP \subseteq \mathrm{NPSPACE} = \mathrm{PSPACE}$$

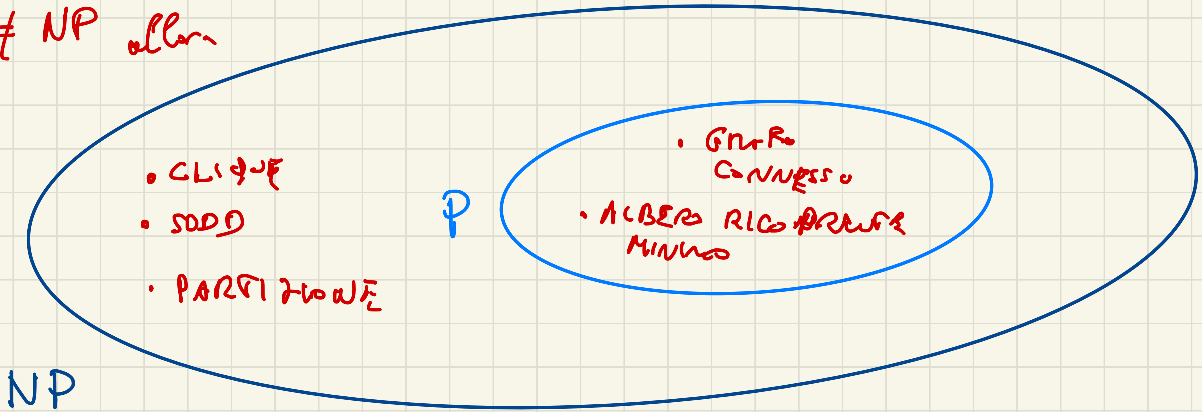
$$P \subseteq \mathrm{PSPACE} \subseteq \mathrm{EXPTIME}$$

$$\downarrow$$

$$P \subseteq NP \subseteq \mathrm{PSPACE} \subseteq \mathrm{EXPTIME}$$

$P \neq NP$?

se $P \neq NP$ allora



Problemi NP-completi → problemi "più difficili" da NP

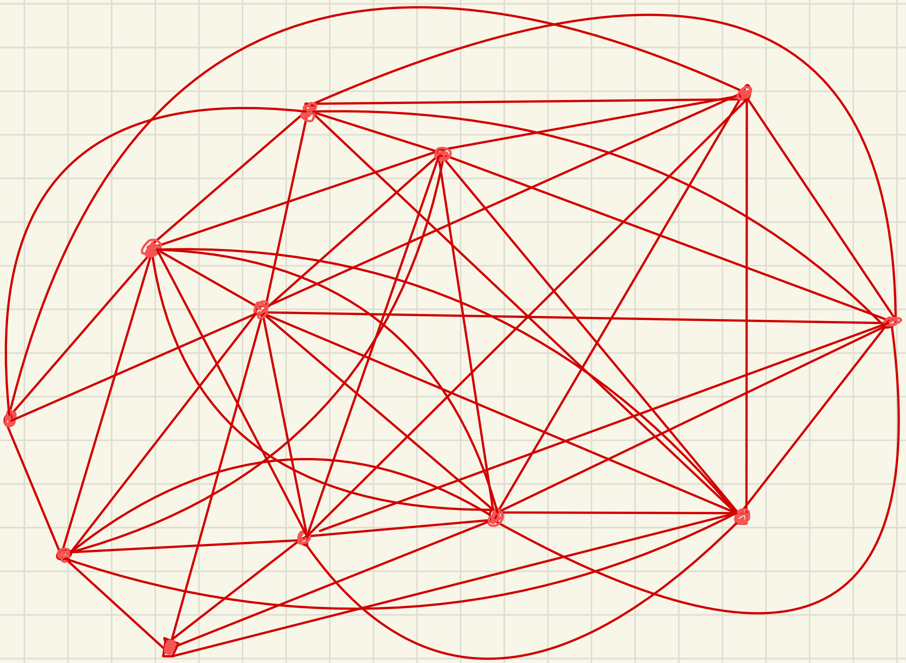
NP-completezza

Problema CLIQUE

Istanza Grafo non orientato $G=(V,E)$

Intero $K \geq 0$

Questione $\exists$ sottografo completo di G con K vertici?



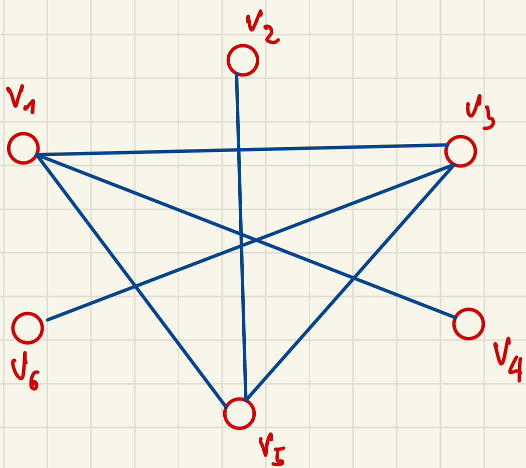
Problema INSIEME INDIPENDENDENTE

Istanza Grafo non orientato $G=(V,E)$

Intero $K \geq 0$

Questione $\exists V' \subseteq V$ con $\# V' = K$ t.c. i vertici in V' non hanno archi che li collegano direttamente?

(cioè se $u, v \in V'$ e $u \neq v$ allora $(u,v) \notin E$)



da CLIQUE a INSIEME INDIPENDENTE

Problema CLIQUE

Istanza: Grafo non orientato $G=(V,E)$

Intero $K \geq 0$

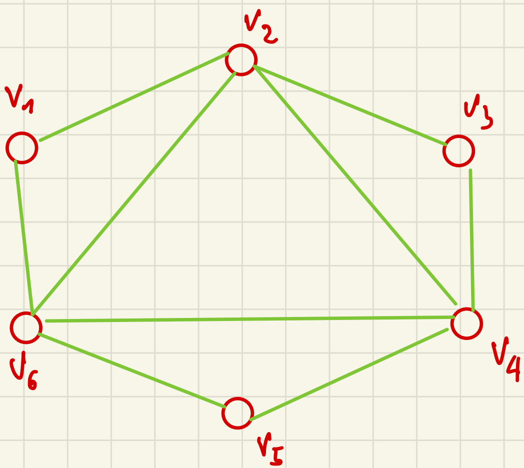
Questione: $\exists$ sottografo completo di G con K vertici?

Problema INSIEME INDIPENDENDENTE

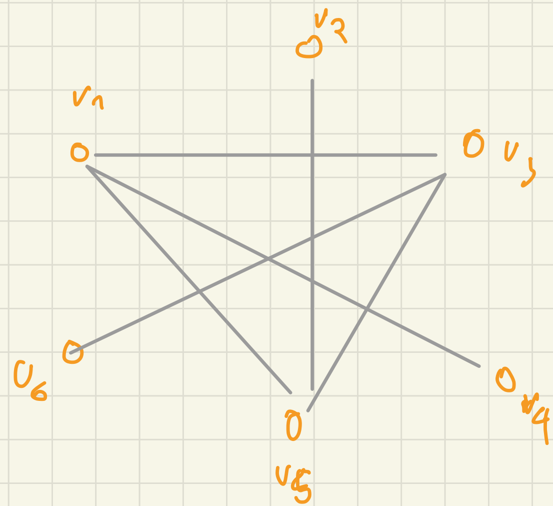
Istanza: Grafo non orientato $G=(V,E)$

Intero $K \geq 0$

Questione: $\exists V' \subseteq V$ con $\# V' = K$ t.c. i vertici in V' non hanno archi che li collegano direttamente?



$K = 3$



$k = 3$

da SODD a CLIQUE

Problema SODDISFACIBILITA' (SODD)

<u>Istanza</u> Formula booleana ϕ in forma normale congiuntiva con insieme di variabili V

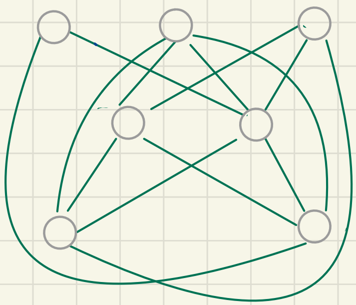
<u>Questione</u> $\exists$ un assegnamento alle variabili in V che rende vera ϕ, cioè t.c. $\phi(f)=1$?

Problema CLIQUE

<u>Istanza</u> Grafo non orientato $G=(V,E)$
Intero $K \geq 0$

<u>Questione</u> $\exists$ sottografo completo di G con K vertici?

$$\phi = (x \vee \bar{y} \vee z) \wedge (\bar{x} \vee \bar{y}) \wedge (\bar{x} \vee \bar{z})$$



$K = 3$

da SODD a CLIQUE

ϕ formula in F.N. congiuntiva

$$\Downarrow$$

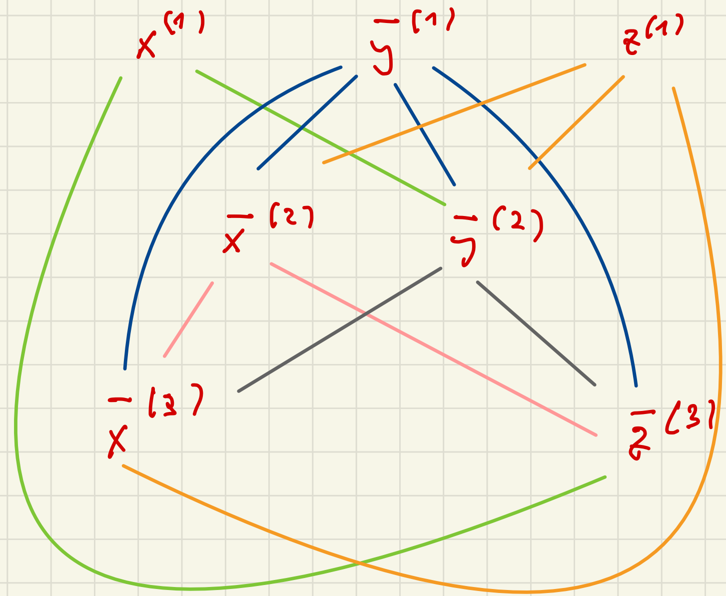
$$\phi = (x \vee \bar{y} \vee z) \wedge (\bar{x} \vee \bar{y}) \wedge (\bar{x} \vee \bar{z})$$

- $G = (V, E)$

  $V = \{$ occorrenze di letterali in $\phi \}$

  $(u, v) \in E$ sse

  u e v sono in clausole differenti e non sono uno il negato dell'altro

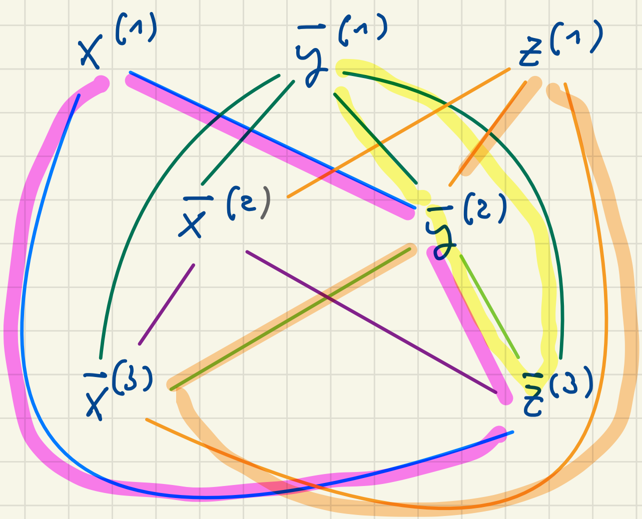


$k = 3$

- $K = \#$clausole di G

Esempio

$$\phi = (x \lor \bar{y} \lor z) \land (\bar{x} \lor \bar{y}) \land (\bar{x} \lor \bar{z})$$



$K = 3$

$\bar{y}^{(2)}$ $\bar{y}^{(1)}$ $\bar{z}^{(3)}$ — $\left.\begin{matrix} x=0 \\ x=1 \end{matrix}\right\}$ $y=0$ $z=0$

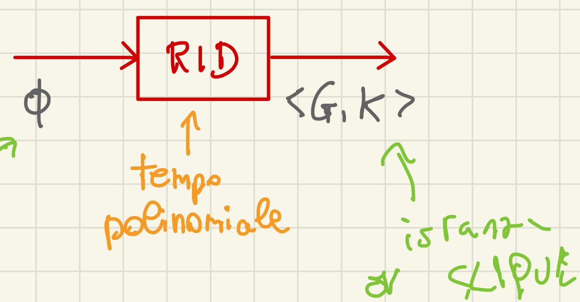
x^{1} $\bar{y}^{(2)}$ $\bar{z}^{(3)}$ — $x=1$ $y=0$ $z=0$

$\bar{x}^{(3)}$ $\bar{y}^{(2)}$ z^{1} — $x=0$ $y=0$ $z=1$

da SODD a CLIQUE in altre parole:

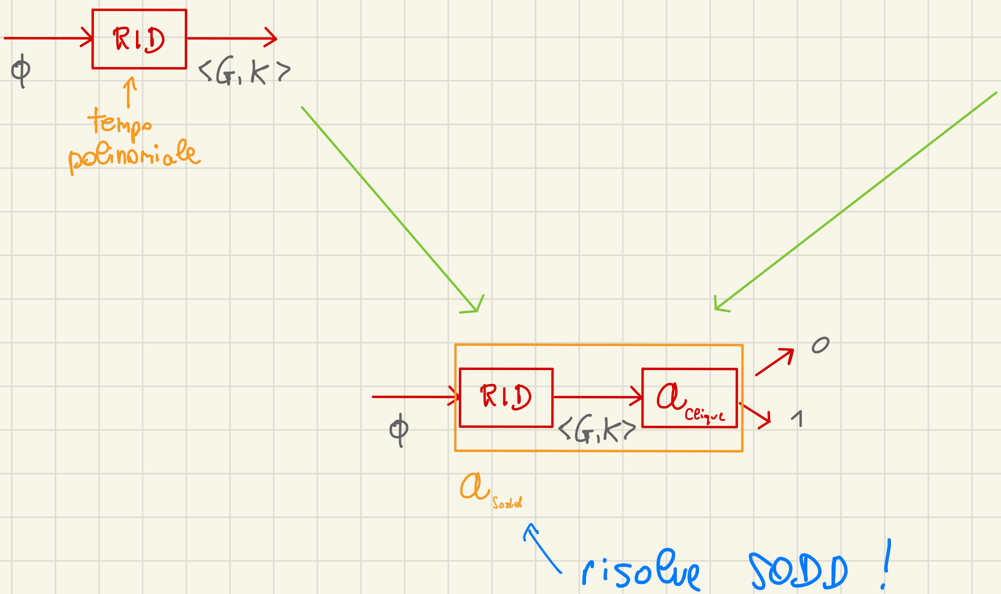
- $\exists$ algoritmo RID che data in ingresso una formula ϕ in forma normale congiuntiva, in TEMPO POLINOMIALE restituisce una coppia $\langle \underset{\text{grafo}}{G}, \underset{\text{intero}}{K} \rangle$ t.c.

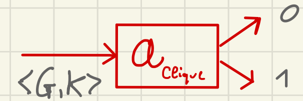
ϕ è soddisfacibile sse G contiene una clique di K vertici



da SODD a CLIQUE

ALGORITMO che risolve CLIQUE





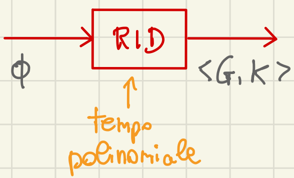
ALGORITMO a_{sodd} (ϕ formula) $\rightarrow$ boolean

$\langle G, K \rangle \leftarrow RID(\phi)$

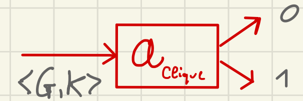
$r \leftarrow a_{clique}(\langle G, K \rangle)$

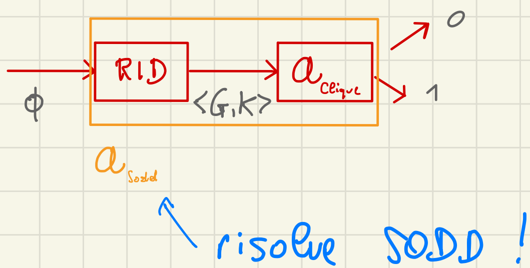
RETURN r

da SODD a CLIQUE



ALGORITMO che risolve CLIQUE





ALGORITMO a_{Sodd} (ϕ formula) $\to$ boolean
  $\langle G, K\rangle \leftarrow RID(\phi)$
  $r \leftarrow a_{Clique}(\langle G, K\rangle)$
  RETURN r

Inoltre:

Se a_{Clique} lavora in tempo polinomiale
anche a_{Sodd} lavora in tempo polinomiale $\Big] \to$ Se CLIQUE $\in P$ allora SODD $\in P$

RIDUCIBILITÀ TRA PROBLEMI

$\Pi_1 : I_1 \rightarrow \{0, 1\}$

$\Pi_2 : I_2 \rightarrow \{0, 1\}$

problemi di decisione

Π_1 è riducibile a Π_2 se $\exists\, f : I_1 \rightarrow I_2$

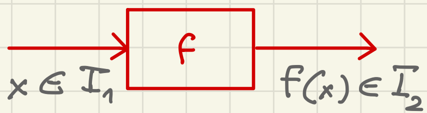
t.c. $\forall x \in I_1 \quad \Pi_1(x) = \Pi_2(f(x))$

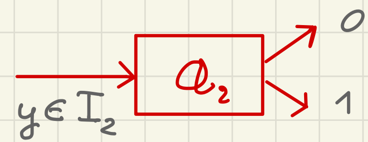
$$x \in I_1 \longrightarrow \boxed{f} \longrightarrow f(x) \in I_2$$

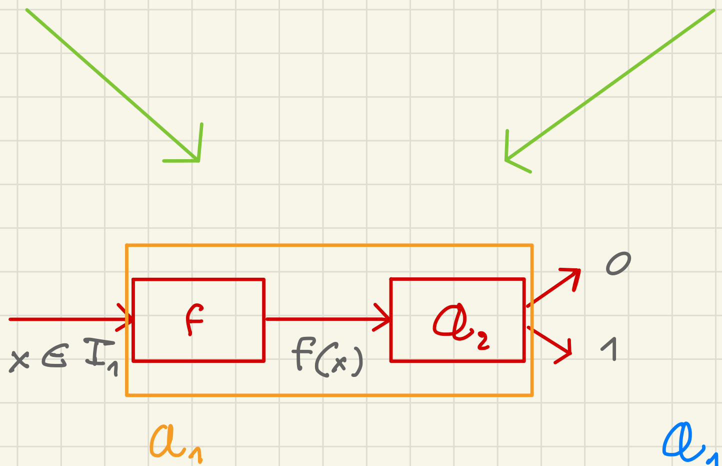
RIDUCIBILITA' TRA PROBLEMI

Supponiamo:

ⓐ Π_1 riducibile a Π_2 ⓑ $\exists$ algoritmo $\mathcal{A}_2$ per Π_2







ALGORITMO $\mathcal{A}_1$ $(x \in I_1) \rightarrow$ boolean
$y \leftarrow f(x)$
$r \leftarrow \mathcal{A}_2(y)$
RETURN r

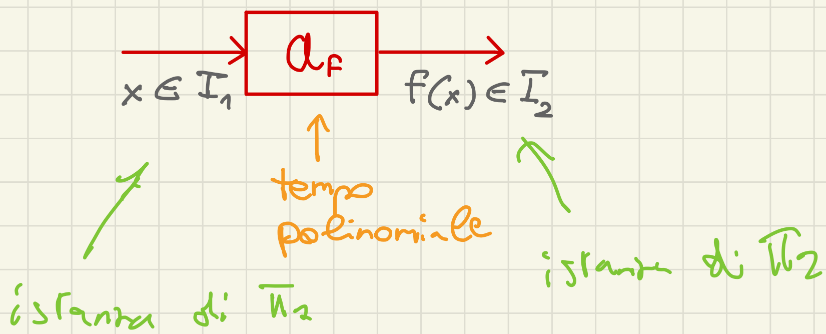
$\mathcal{A}_1$ è un algoritmo per Π_1

RIDUCIBILITA' POLINOMIALE

Π_1 è riducibile POLINOMIALMENTE a Π_2 $\quad (\Pi_1 \leq_p \Pi_2)$

se $\exists$ riduzione $f: I_1 \to I_2$ calcolabile

da un algoritmo $\mathcal{A}_f$ in tempo polinomiale

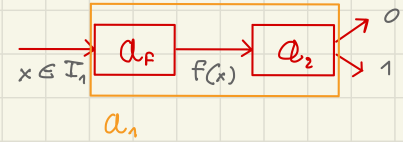


ALGORITMO $\mathcal{A}_1(x \in I_1) \to$ boolean

$y \leftarrow \mathcal{A}_f(x)$ $\longrightarrow$ tempo $O(p(|x|))$

$r \leftarrow \mathcal{A}_2(y)$ $\longrightarrow$ tempo $O(q(|y|))$

RETURN r



$\mathcal{A}_f$ tempo $p(n)$ (polinomio)

$\mathcal{A}_2$ tempo $q(n)$ (polinomio)

Se $\mathcal{A}_f$ e $\mathcal{A}_2$ lavorano in tempo polinomiale,
in quanto tempo lavora $\mathcal{A}_1$?

tempo totale $O(p(|x|) + q(|y|)) = O(p(|x|) + q(p(|x|)))$

In tempo $p(n)$ produco output di lunghezza $O(p(n))$

$|y| = O(p(|x|))$

poly

Comp di poly è un poly

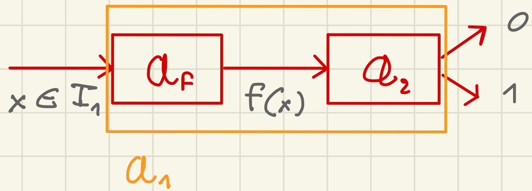
poly + poly $\to$ poly

$= O(\text{poly}(|x|))$

RIDUCIBILITÀ POLINOMIALE

Proprietà fondamentali

1. Se $\Pi_1 \leq_p \Pi_2$ e $\Pi_2 \in P$ allora $\Pi_1 \in P$



<u>Esempio</u> Π_1 = SODD Π_2 = CLIQUE

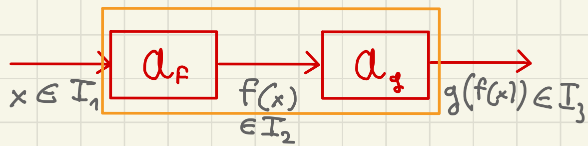
SODD $\leq_p$ CLIQUE

Se CLIQUE $\in P$ allora SODD $\in P$

RIDUCIBILITA' POLINOMIALE

Proprietà fondamentali

2. Se $\Pi_1 \leq_p \Pi_2$ e $\Pi_2 \leq_p \Pi_3$ allora $\Pi_1 \leq_p \Pi_3$



Esempio

SODD $\leq_p$ CLIQUE

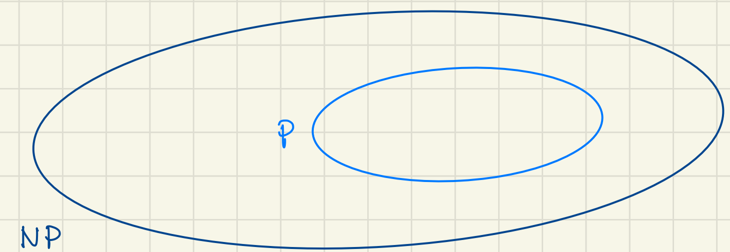
CLIQUE $\leq_p$ INSIEME INDIPENDENTE

$\longrightarrow$ SODD $\leq_p$ INSIEME INDIPENDENTE

PROBLEMI NP-HARD e NP-COMPLETI

- Π è NP-HARD o NP-difficile se $\forall \Pi' \in NP : \Pi' \leq_p \Pi$ (Π' è riducibile polinomialmente a Π)

- Π è NP-completo se:
  - Π è NP-Hard e
  - $\Pi \in NP$

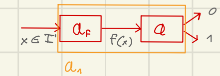


$\Downarrow$

Π è un problema "più difficile" tra tutti i problemi in NP

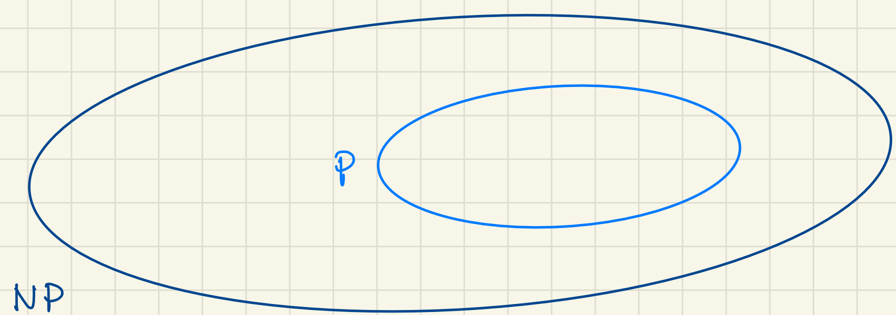
PROBLEMI NP-completi

① Se $\Pi' \leq_p \Pi$ e $\Pi \in P$ allora $\Pi' \in P$



<u>Teorema</u> Sia Π un problema NP-completo.

- Se $\Pi \notin P$ allora $P \neq NP$
- Se $\Pi \in P$ allora $P = NP$



TEOREMA DI COOK

SODD è NP-completo

Dim

$\boxed{1}$ SODD $\in NP$

$\boxed{2}$ $\forall \Pi' \in NP \quad \forall \Pi' \leq_p$ SODD

CLIQUE è NP-completo

Proprietà

$\Pi_1 \leq_p \Pi_2$ e $\Pi_2 \leq_p \Pi_3$ allora $\Pi_1 \leq_p \Pi_3$

Dim

$\boxed{1}$ CLIQUE $\in$ NP

$\boxed{2}$ Abbiamo visto che SODD $\leq_p$ CLIQUE

Poiché SODD è NP-completo: $\forall \Pi' \in P$ $\Pi' \leq_p$ SODD

$\Pi' \leq_p$ CLIQUE $\rightarrow$ CLIQUE e' NP-Hard

CLIQUE NP-completo

In generale

Se Π è NP-completo e $\Pi \leq_p \tilde{\Pi}$ con $\tilde{\Pi} \in NP$

allora $\tilde{\Pi}$ è NP-completo

Se $P \neq NP$: