PROBLEMA DELL'INSIEME INDIPENDENTE (INDEPENDENT SET)

INPUT: Grafo non orientato G

SOL. AMMISSIBILE: $X \subseteq V_G$ t.c. $\binom{X}{2} \cap E = \emptyset$

COSTO: $|X|$

TIPO: MAX

Teorema: Se $P \neq NP$, non è possibile approssimare INDEPENDENT SET a meno di $(2-\varepsilon)$, per qualche $\varepsilon > 0$.

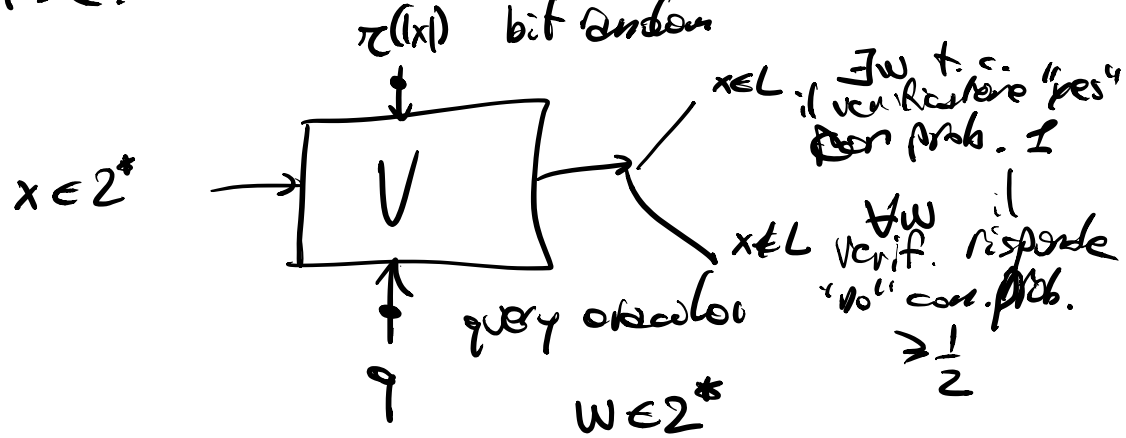
$L \subseteq 2^*$ t.c. L è NP-completo

$\Rightarrow$ per il teorema PCP esiste un verificatore

$V \in PCP[r(n), q]$ con

$r(n) \in O(\lg n)$ e $q \in \mathbb{N}$

t.c. V accetta L



$x \in 2^*$ fissato

- Simulo V per ogni $R \in 2^{r(|x|)}$ e per ogni sequenza $\in 2^q$ di risposte.

$$\#\text{simulazioni} = 2^{r(|x|)} \cdot 2^q = 2^{O(\lg |x|)} \cdot 2^q \in \text{Poly}$$

- Tengo traccia delle sole simulazioni accettanti ("yes").

$$x = \Big(\underset{2^{r(|x|)}}{R},\ \underbrace{\{i_1^R : v_1,\ i_2^R : v_2,\ \dots,\ i_q^R : v_q\}}_{\text{query}}\Big)$$

$$x' = \Big(R',\ \{i_1^{R'} : v_1',\ i_2^{R'} : v_2',\ \dots,\ i_q^{R'} : v_q'\}\Big)$$

c'è un lato se

$R = R'$ $\vee$ $\{i_1^R : v_1, \dots\}$ è incomp. con $\{i_1^{R'} : v_1', \dots\}$

$\rightarrow \{3 : 1,\ \underline{15} : 0,\ 17 : 1\}$

$\{\underline{15} : 1,\ 18 : 0,\ 19 : 0\}$

<u>NON</u> c'è un arco se

$R \neq R'$ $\wedge$ le seq. domande/risp. sono compatibili

$$\begin{array}{cccccccccccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 \\ \end{array}$$

$$W = 1\,0\,1\,1\,0\,1\,0\,0\,0\,0\,1\,0\,1\,0\,0\,0\,1 \dots$$

$x \in 2^* \longrightarrow$ **SIMULATORE (V)** in tempo polinomiale $\longrightarrow G_x \longrightarrow$ **A** che α-approssima di MAXSET

$\downarrow$

un insieme indip. di cardinalità $|S^*_{G_x}|$

$$|S_{G_x}| \geq \frac{|S^*_{G_x}|}{\alpha}$$

se fosse $\alpha < 2$

$$x \in L \Rightarrow |S^*_{G_x}| \underset{\text{Fatto 1}}{\geq} 2^{r(|x|)}$$

$$\Rightarrow |S_{G_x}| \geq \frac{2^{r(|x|)}}{\alpha}$$

$$x \notin L \Rightarrow |S^*_{G_x}| \underset{\text{Fatto 2}}{\leq} \frac{2^{r(|x|)}}{2}$$

$$\Rightarrow |S_{G_x}| \leq \frac{2^{r(|x|)}}{2}$$

Fatto 1: Se $x \in L$, G_x ha un insieme indipendente di card. $\geq 2^{r(|x|)}$

Dim: $\exists w \in 2^*$ qualunque che fa accettare V sia R.

$(R, \ldots)$ $(R', \ldots)$ $\ldots$ $(R'', \ldots)$

compatibili con w

$\geq 2^{r(|x|)}$

$\square$

Fatto 2: Se $x \notin L$, gli insiemi indipendenti di G_x hanno cardinalità $\leq 2^{r(|x|)-1}$.

Dim: Per assurdo. Supponete che ci sia un insieme $S \subseteq V_{G_x}$ indipendente. $|S| > 2^{r(|x|)-1}$ $\Rightarrow \exists w_S$ per cui tutte le simulazioni in S si verificherebbero per più di $\frac{2^{r(|x|)}}{2}$ $\square$

ho un'accettazione su w_S.

ALGORITMICA

ALGORITMICA END-TO-END

- Tecniche di progettazione
- Studio di proprietà di algoritmi
  - complessità
  - approssimazione
  - probabilistiche
- Studio di proprietà di problemi
- complessità strutturale

SVILUPPO DI STRUTTURE DATI

STRUTTURE DATI

- ADT (Abstract Data Type)
- Implementazione

Stack<T>

- Stack<T>()
- void pop() throws NoSuchElement Exception
- T top()
- void push(T x)

1) $\forall S \quad \forall x \in T$

$$S.push(x).top() = x$$

2) $\forall S$

$$S.push(x).pop() = S$$

...

INFORMATION-THEORETICAL LOWER BOUND

- Struttura dati che deve assumere v stati diversi

D [STACK di valori $0, \dots, 9$ di altezza ≤ 5]

$$v = \underbrace{1 + 10 + 10^2 + \dots + 10^5}_{111111} =$$

$$\lceil \log_2 v \rceil \text{ bit}$$

k bit con $k < \log_2 v$

$$2^k < 2^{\log_2 v} = v$$

$$\log_2 v = 16.76\dots$$

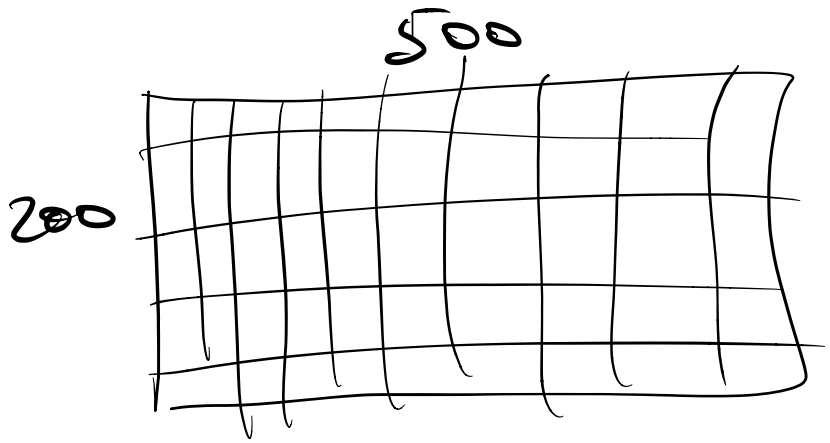
$$2^{16} = 65536$$

$x_1 \quad x_2 \quad \dots \quad x_v$

$$\frac{x_1 + x_2 + \dots + x_v}{v} \geq \log_2 v$$

<u>Thm</u>: Una struttura dati con v valori possibili che usi $x_1, x_2, \dots, x_v$ bit per rappresentare tali valori soddisfa

$$\frac{\sum_{i=1}^{v} x_i}{v} \geq \log_2 v .$$



boldjpeg

BIANCO E NERO

$$2^{500 \cdot 200} = 2^{100000}$$

$$x_1, x_2, \ldots, x_n$$

STRUTTURE SPACE-CONSCIOUS

Considerate un ADT in cui Z_n sia l'information-theoretical lower bound per la taglia n.

Sia D_n lo spazio medio occupato da una certa implementazione.

Ovviamente:

$$D_n \geq Z_n.$$

La struttura si chiama

1) <u>implicita</u> se $D_n = Z_n + O(1)$
2) <u>succinta</u> se $D_n = Z_n + o(Z_n)$
3) <u>compatta</u> se $D_n = O(Z_n)$

1) $Z_n + 3$ 2) $Z_n + \lg Z_n$ 3) $2 Z_n$

a patto che le primitive abbiano

gli stessi tempi di accesso della versione non compressa.