# PASCAL MINIMO NON DETERMINISTICO

INPUT:

$X = x_1, \dots, x_n$ MEMORIZZATA IN $N(N[1] = x_1, \dots, N[n] = x_n)$, $x \in \Sigma^*$

COSTANTI:

$P = \{P_1, \dots, P_k\}$ CON $P_i = \langle q_{i1}, s_{i1}, s_{i2}, q_{i2}, m_i \rangle, q_0, q_A, q_R$

$q \leftarrow q_0$; $c \leftarrow 1$; $pc \leftarrow 1$; $uc \leftarrow n$

WHILE $(q \neq q_A \vee q \neq q_R)$ DO BEGIN:

$\Psi = \{p_i \in P : p_i = \langle q, N[c], \_, \_, \_ \rangle\}$ → $\Psi$ È L'INSIEME DELLE MULTIQUINTUPLE DI $P_i$

↳ $|\Psi|$ È COSTANTE IN QUANTO È AL MASSIMO UGUALE AL GRADO DI NON DETERMINISMO

IF $(\Psi \neq \emptyset)$ THEN BEGIN:

SCEGLI $P_i = \langle q_i, N[c], s_{i2}, q_{i2}, m_i \rangle \in \Psi_i$

$N[c] \leftarrow s_{i2}$, $q \leftarrow q_{i2}$, $c \leftarrow c + m$

IF $(c < pc)$ THEN ....

IF $(c > uc)$ THEN.....

END IF

END WHILE

RETURN $q$

## OSS

1. L'ISTRUZIONE SCEGLI È L'ISTRUZIONE CHE IMPLEMENTA IL NON DETERMINISMO

2. L'ALGORITMO È BASATO SUL MODELLO DEL GENIO CHE SCEGLIE LA QUINTUPLA DA ESEGUIRE

3. POSSIAMO FIDARCI DEL GENIO SOLO SE AL TERMINE DELL'ALGORITMO CI PORTA IN $q_A$

4) USANDO LA DEFINIZIONE DI NP COME MACCHINA ACCETTANTE A NOI NON IMPORTA DI QUALSIASI RISULTATO CHE NON SIA $q_A$

5) SE $\exists$ UNA COMPUTAZIONE CHE PORTA IN $q_A$ ALLORA $\exists$ UN GENIO CHE ME LA SELEZIONA

## PROBLEMA 3-SAT

$$\mathcal{I}_{3\text{-SAT}} = \{ \langle X, f \rangle : f = C_1 \wedge C_2 \wedge \ldots \wedge C_m ; \forall i \in [0, m] \; [ C_i = l_{i1} \vee l_{i2} \vee l_{i3}, \forall_{ij} \; l_{ij} \in X \vee \neg l_{ij} \in X ] \}$$

$$S_{3\text{-SAT}}(X, f) = \{ a : X \to \{\text{VERO}, \text{FALSO}\} \}$$

$$\pi_{3\text{-SAT}}(X, f, S(X, f)) = \exists a \in S_{3\text{-SAT}}(X, f) : f(a(X)) = \text{VERO}$$

## ALGORITMO 3-SAT

INPUT: $X$ ; $f = C_1 \wedge C_2 \wedge \ldots \wedge C_m$

$q \leftarrow q_0$; $c \leftarrow 1$; $pc \leftarrow 1$; $uc \leftarrow n$

FOR $(i \leftarrow 1;\ i \leq |X|;\ i \leftarrow i+1)$ DO

  SCEGLI $a(x_i)$ IN $\{\text{VERO}, \text{FALSO}\}$

FOR $(i \leftarrow 1;\ i \leq |X|;\ i \leftarrow i+1)$ DO

  SOSTITUISCI $x_i$ CON $a(x_i)$ IN $f$

IF $(f = \text{VERO})$ THEN

  $q \leftarrow q_A$

RETURN $q$

$O(|f|)$

FASE DI SCELTA → ESEGUITA IN $O(|X|)$

PUO' ESSERE FATTA SOLO FRA UN NUMERO COSTANTE DI POSSIBILITÀ

FASE DI VERIFICA → ESEGUITA IN $O(|X||f|)$

SI CONTROLLA SE L'ISTANZA CREATA NELLA FASE DI SCELTA PORTI IN $q_A$

## CLASSE DI COMPLESSITÀ

L'ALGORITMO ACCETTA 3-SAT IN TEMPO NON DETERMINISTICO $O(|X||f|)$

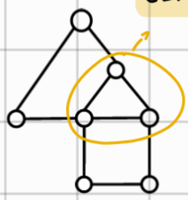
  3-SAT $\in$ NP

# PROBLEMA DEL SOTTOINSIEME CLIQUE

PER OGNI COPPIA DI NODI ESISTE UN'ARCO

SOTTOINSIEME DI NODI CONNESSI DI CARDINALITÀ ALMENO $K$

CLIQUE DI $K=3$



$$\mathcal{I}_{CL} = \{ \langle G(V,E), K \rangle : G \text{ GRAFO}, K \in \mathbb{N} \}$$

$$S_{CL}(G,K) = \{ V' \subseteq V \}$$

$$\pi_{CL}(G,K,S(G,K)) = \exists\, V' \in S(G,K) : \forall v_i, v_j \in V' \left[ (v_i, v_j) \in E \right] \wedge |V'| \geq K$$

## ALGORITMO CLIQUE

INPUT: $G=(V,E)$, $K \in \mathbb{N}$

$V' \leftarrow \emptyset$

FOR EACH $(v \in V)$:

  SCEGLI SE $v \in V'$

FASE DI SCELTA → ESEGUITA IN $O(|V|)$

ESEGUITA A LIVELLO DI $S_{CL}$

CLIQUE ← TRUE

IF $(|V'| < K)$ THEN → $O(|V|)$

  CLIQUE ← FALSE

FOR EACH $(v \in V')$ → $O(|V|^2)$

  FOR EACH $(u \in V')$

    IF $((u \neq v) \wedge (u,v) \notin E)$ THEN: → $O(|E|)$

      CLIQUE ← FALSE

RETURN CLIQUE

FASE DI VERIFICA → ESEGUITA IN $O(|V|^2|E|)$

ESEGUITA A LIVELLO DI $\pi_{CL}$

## CLASSE DI COMPLESSITÀ

L'ALGORITMO ACCETTA IL PROBLEMA CLIQUE IN TEMPO NON DETERMINISTICO $O(|V|^2|E|)$

  IL PROBLEMA CLIQUE $\in NP$

# Problema del ciclo Hamiltoniano

Un ciclo Hamiltoniano è un ciclo che passa per ogni nodo del grafo una ed una volta sola

$$I_{HC} = \{ \langle G=(V,E) \rangle : |V| = n \}$$

$$S_{HC}(G) = \{ P : \langle v_1, v_2, \dots, v_n \rangle : \forall i \, [v_i \in V] \}$$ → Cammino comprendente tutti i nodi del grafo

$$\pi_{HC}(G, S(G)) = \exists P \in S_{HC}(G) : \forall i \neq j \, [v_i \neq v_j] \wedge \forall i = 1 \dots n \, [(v_i, v_{i+1}) \in E \wedge (v_n, v_1) \in E]$$

↳ $\exists P \in S(G) : \pi(G, S(G))$

## Algoritmo cammino Hamiltoniano

```
INPUT: G=(V,E)

FOR (i←1; i≤n; i←i+1) DO
    FOR (v∈V) DO
        SCEGLI SE v=v_i O v≠v_i

CICLE←TRUE
FOR (i←1; i≤n; i←i+1) DO
    FOR (j←1; i≤n; i←i+1) DO
        IF (v_i≠v_j) THEN CYCLE←FALSE
FOR (i←1; i≤n; i←i+1) DO
    IF ((v_i, v_i+1)∉E) THEN CYCLE←FALSE
IF ((v_n, v_1)∈E ∧ CYCLE=TRUE) THEN
    RETURN CYCLE
```

Serve a scorrere le posizioni di P

Fase di scelta → $O(n^2)$

Fase di verifica → $O(n^2)$

## Classe di complessità

L'algoritmo accetta il problema del cammino Hamiltoniano in tempo ND $O(n^2)$

IL PROBLEMA DEL CAMMINO HAMILTONIANO ∈ NP

# DEFINIZIONE ALTERNATIVA DI NP → PROVVISORIA

ISTANZA SI

Y SOSTITUISCE S(x)

DATO $\Gamma$ : $\pi_\Gamma(X, S_\Gamma(X)) = \exists\, Y \in S_\Gamma(X) : \eta_\Gamma(Y, X) = \text{VERO}$

SOLUZIONE POSSIBILE

$\eta(x,y)$ E' VERIFICABILE IN TEMPO DETERMINISTICO POLINOMIALE $\wedge$ Y E' COSTRUITO IN TEMPO NON DETERMINISTICO POLINOMIALE $\Rightarrow [\Gamma \in NP]$

## ALGORITMO NON DET. STAMPO

INPUT: $x \in \mathcal{I}_\Gamma$

COSTRUISCI NON DETERMINISTICAMENTE $Y \in S_\Gamma(x)$

VERIFICO DETERMINISTICAMENTE $\eta(Y)$