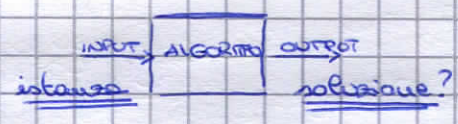


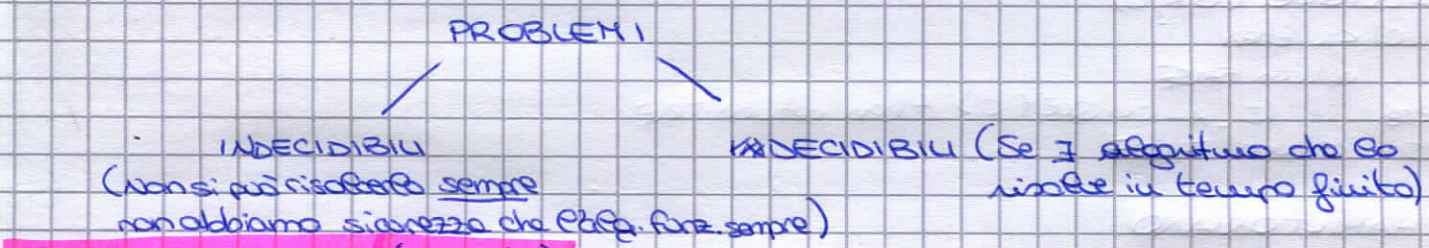
PROBLEMA =



$I$  = istanze  
 $S$  = soluzioni  
 $P \subseteq I \times S$

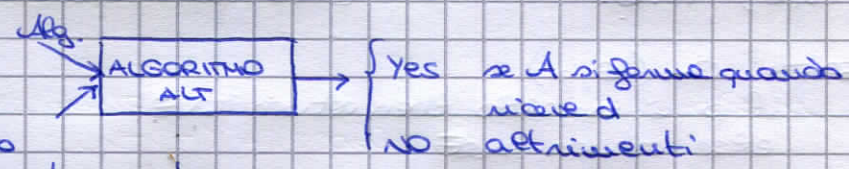
Un problema è una relazione tra istanze e soluzioni (è sottoinsieme del prodotto cartesiano di istanze e soluzioni)

Un algoritmo risolve un problema se data <sup>un'istanza</sup> in input restituisce una soluzione



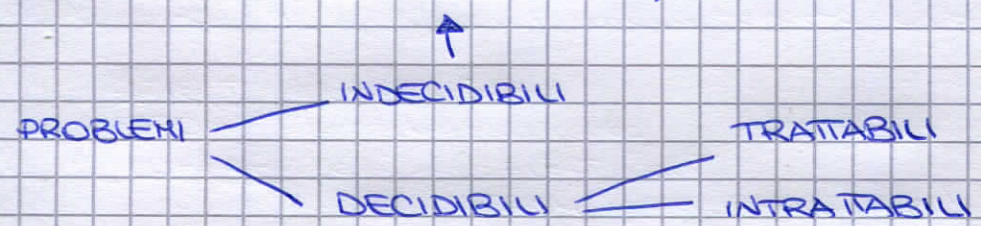
PROBLEMA DELL'ALT (fermato)

INDECIDIBILE



L'algoritmo di Alt misura il comportamento degli algoritmi in ingresso, per questo è indecidibile

NON TRATTIAMO GLI INDECIDIBILI, TRATTIAMO I DECIDIBILI



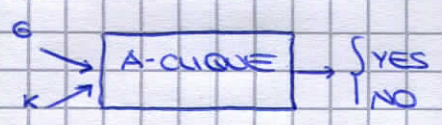
EFFICIENZA (INTRATTABILITÀ) ha a che fare con la polinomialità. Un problema è trattabile se esiste un algoritmo con complessità  $O(n^k)$

" INTRATTABILE:  $O(2^n)$  oppure esponenziale  $O(k^n)$

PROBLEMI DECISIONALI: Bisogna rispondere a una certa domanda binaria (sì oppure no)  
 Per esempio il problema della clique (≠ alg. clique max)

PROBLEMI DI OTTIMIZZAZIONE: Minimizzare / massimizzare una funzione data

CLIQUE (= decisionale)



Istanze: grafi G non orientati; interi  
 Soluzioni: YES/NO

MAX-CLIQUE (= ottimizzazione)

Istanze: grafi G non orientati; Soluzione: clique massima



MAX-CLIQUE(G)  $O(n^r)$  dove  $r = \text{costante}$

```
1. Clique(G, k)
2.   C ← MaxClique(G)
3.   if |C| ≥ k then
4.     return TRUE
5.   else
6.     return FALSE
```

$P = \{P \mid P \text{ è un problema decisionale risolvibile in tempo polinomiale}\}$

$NP = \{P \mid P \text{ è un problema decisionale verificabile in tempo polinomiale}\}$

```
VERIFY-CLIQUE(G, R, C)
  if |C| ≤ k then
    return FALSE
  else if IS_A-CLIQUE(C) then return TRUE
  else return FALSE
```

GRAFO HAMILTONIANO

$G = (V, E)$  n.o.

Ciclo Hamiltoniano  $\equiv$  ciclo che passa attraverso tutti i vertici di  $G$ .

```
VERIFY-GH(G, p)
  if p è un ciclo then return TRUE
  else return FALSE
```

- 1)  $P \subseteq NP$  ?
- 2)  $NP \subseteq P$  ?
- 3)  $NP \cap P = \emptyset$  ? X
- 4)  $NP \cap P \neq \emptyset$  ? X