Automi e Linguaggi Formali

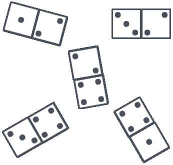
Parte 19 – La classe NP

Davide Bresolin
Ultimo aggiornamento: 19 maggio 2021

Università degli Studi di Padova

Disponete in fila le tessere del domino che vi sono state consegnate:

1. in modo da usare tutte le tessere

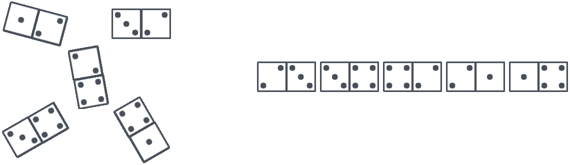


Giochiamo a Domino[1]

Università degli Studi di Padova

Disponete in fila le tessere del domino che vi sono state consegnate:

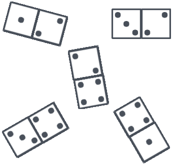
1. in modo da usare tutte le tessere



Giochiamo a Domino[1]

Disponete in fila le tessere del domino che vi sono state consegnate:

1. in modo da usare tutte le tessere



Domanda:

È un problema facile o difficile da risolvere?

Come mostrare che un problema è facile?

Definizione

Un problema è trattabile (facile) se esiste un algoritmo efficiente per risolverlo.

- Gli algoritmi efficienti sono algoritmi con complessità polinomiale:
  - il loro tempo di esecuzione è $O(n^k)$ per qualche costante k.
- Avere complessità polinomiale è una condizione minima per considerare un algoritmo efficiente
- Un algoritmo con complessità più che polinomiale (p.es. esponenziale) è un algoritmo non efficiente perché non è scalabile.

Mostriamo che Domino[1] è trattabile

Università degli Studi di Padova

Obiettivo

Trovare un algoritmo polinomiale per Domino[1]

1. Formulazione del problema in termini di linguaggio
2. Definizione di una Macchina di Turing che lo decide
3. Analisi di complessità della macchina di Turing (o dell'algoritmo)

Un linguaggio e una riduzione

$$D_1 = \{\langle B\rangle \mid B \text{ è un insieme di tessere del domino, ed esiste un allineamento che usa tutte le tessere}\}$$

- Usiamo una riduzione mediante funzione per trovare l'algoritmo polinomiale
- Riduciamo D_1 ad un problema su grafi ...
- ... per il quale sappiamo che esiste un algoritmo polinomiale

Dalle tessere al grafo

Definition (Grafo)

Un grafo (non orientato) G è una coppia (V, E) dove:

- $V = \{v_1, v_2, \ldots, v_n\}$ è un insieme finito e non vuoto di vertici;
- $E \subseteq \{\{u, v\} \mid u, v \in V\}$ è un insieme di coppie non ordinate, ognuna delle quali corrisponde ad un arco del grafo.

Università degli Studi di Padova

Definition (Grafo)

Un grafo (non orientato) G è una coppia (V, E) dove:

- $V = \{v_1, v_2, \ldots, v_n\}$ è un insieme finito e non vuoto di vertici;
- $E \subseteq \{\{u, v\} \mid u, v \in V\}$ è un insieme di coppie non ordinate, ognuna delle quali corrisponde ad un arco del grafo.

Grafo del domino

- Vertici: i numeri che si trovano sulle tessere
  - $V = \{$⚀, ⚁, ⚂, ⚃$\}$
- Archi: le tessere del domino
  - $E = \{$[⚂⚃], [⚀⚁], [⚃⚁], [⚃⚀], [⚂⚁]$\}$

Domino[1] è un problema su grafi!

- **Cammino Euleriano**: percorso in un grafo che attraversa tutti gli archi una sola volta

Il problema del Cammino Euleriano

$EULER = \{\langle G \rangle \mid G$ è un grafo che possiede un cammino Euleriano$\}$

- *EULER* è un problema classico di teoria dei grafi
- Esistono algoritmi polinomiali per risolverlo

Algoritmo di Fleury

1. Scegliere un vertice con grado dispari (un vertice qualsiasi se tutti pari)
2. Scegliere un arco tale che sua cancellazione non sconnetta il grafo
3. Passare al vertice nell'altra estremità dell'arco scelto
4. Cancellare l'arco dal grafo
5. Ripetere i tre passi precedenti finche non eliminate tutti gli archi

Complessità

Su un grafo con n archi, l'algoritmo di Fleury impiega tempo $O(n^2)$

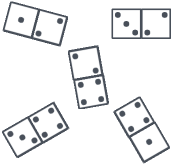
Complessità di Domino[1]

- L'algoritmo di Fleury risolve *EULER* in tempo polinomiale
- La riduzione ci dice che $D_1 \leq_m EULER$
- Quanto tempo serve per risolvere il problema D_1?

Giochiamo a domino[2]

Disponete in fila le tessere del domino che vi sono state consegnate:

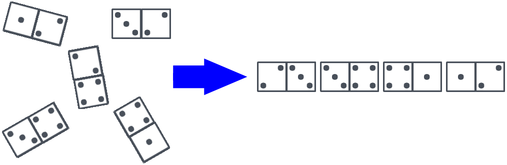
2. in modo che ogni numero compaia esattamente due volte (potete usare meno tessere di quelle che avete).



Giochiamo a domino[2]

Disponete in fila le tessere del domino che vi sono state consegnate:

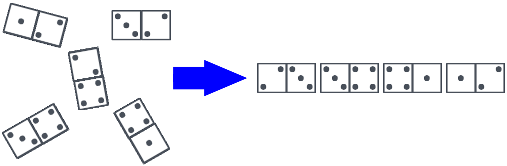
2. in modo che ogni numero compaia esattamente due volte (potete usare meno tessere di quelle che avete).



Giochiamo a domino[2]

Disponete in fila le tessere del domino che vi sono state consegnate:

2. in modo che ogni numero compaia esattamente due volte (potete usare meno tessere di quelle che avete).



Domanda:

È un problema facile o difficile da risolvere?

Una riduzione in senso opposto!

$$D_2 = \{\langle B\rangle \mid B \text{ insieme di tessere del domino, ed esiste allineamento dove ogni numero compare due volte}\}$$

- **Circuito Hamiltoniano**: ciclo nel grafo che attraversa tutti i vertici una sola volta

Il problema del Circuito Hamiltoniano

$$HAMILTON = \{\langle G\rangle \mid G \text{ è un grafo con un circuito Hamiltoniano}\}$$

- Come facciamo a dimostrare che $HAMILTON \leq_m D_2$?

HAMILTON è un problema difficile!

- Il problema del circuito Hamiltoniano è un problema classico di teoria dei grafi
- Un algoritmo polinomiale per risolverlo non è mai stato trovato
- Se qualcuno mi dà una possibile soluzione, è facile verificare se è corretta

Problemi trattabili e problemi intrattabili

- I problemi per i quali esiste un algoritmo polinomiale vengono considerati trattabili
- quelli che richiedono un algoritmo più che polinomiale sono detti intrattabili.
- Sappiamo che ci sono problemi che non possono essere risolti da nessun algoritmo:
  - "Halting Problem" di Turing
- Ci sono problemi che richiedono un tempo esponenziale:
  - il gioco della Torre di Hanoi

Problemi trattabili e problemi intrattabili

- I problemi per i quali esiste un algoritmo polinomiale vengono considerati trattabili
- quelli che richiedono un algoritmo più che polinomiale sono detti intrattabili.
- Sappiamo che ci sono problemi che non possono essere risolti da nessun algoritmo:
  - "Halting Problem" di Turing
- Ci sono problemi che richiedono un tempo esponenziale:
  - il gioco della Torre di Hanoi

Stabilire con precisione qual'è il confine tra problemi trattabili ed intrattabili è piuttosto difficile

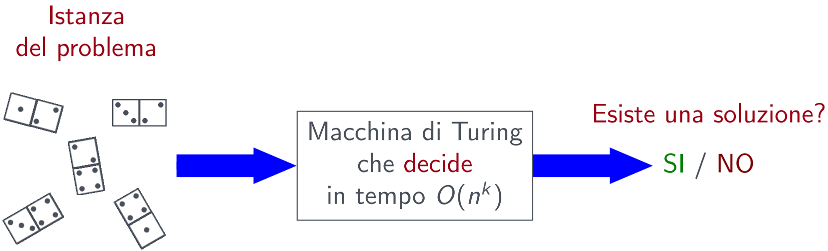
Facili da risolvere

Facili da verificare

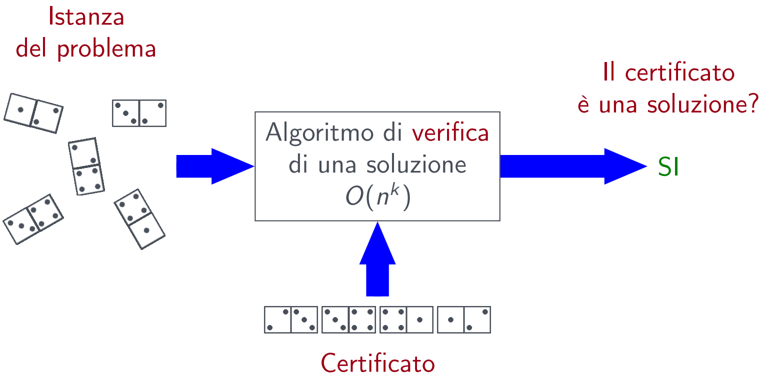
	P
Facili da risolvere	✓
Facili da verificare	✓
Esempi	Domino[1], Euler, ordinamento, ...

	P	NP
Facili da risolvere	✓	?
Facili da verificare	✓	✓
Esempi	Domino[1], Euler, ordinamento, ...	Domino[2], Hamilton, Sudoku, Protein folding, Crittografia, ...

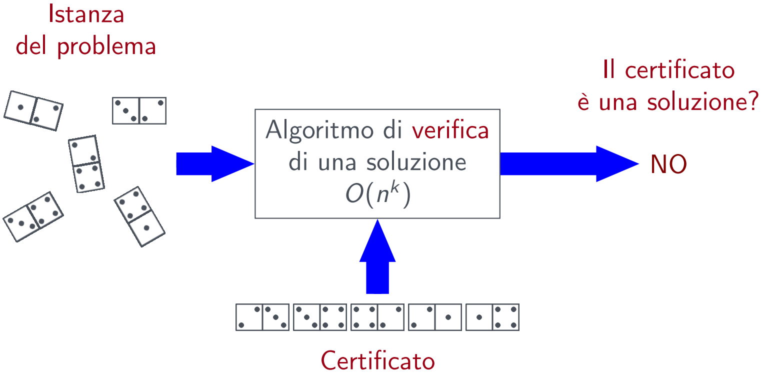
Domino[1] è in P



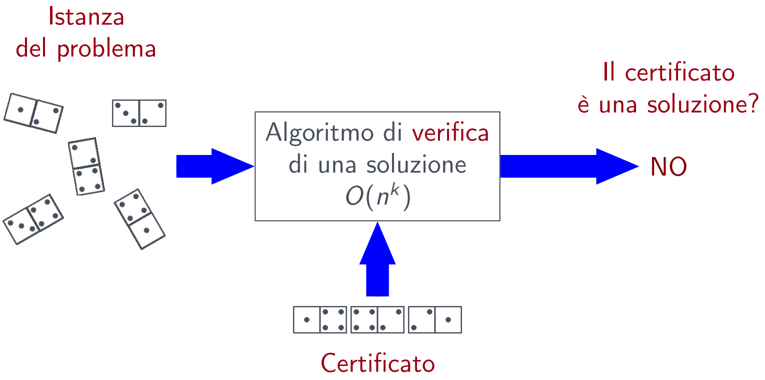
Domino[2] è in NP



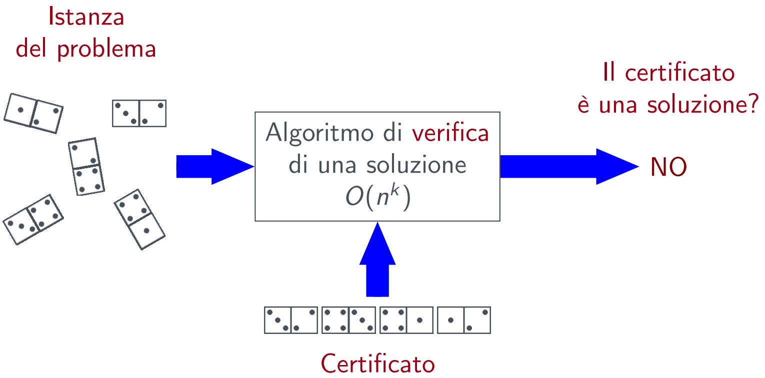
Domino[2] è in NP



Domino[2] è in NP



Domino[2] è in NP



Verificatori

Definition

Un verificatore per un linguaggio A è un algoritmo V tale che

$$A = \{w \mid V \text{ accetta } \langle w, c\rangle \text{ per qualche stringa } c\}$$

- il verificatore usa ulteriori informazioni per stabilire se w appartiene al linguaggio
- questa informazione è il certificato c

- **P** è la classe dei linguaggi che possono essere decisi da una macchina di Turing deterministica che impiega tempo polinomiale.

- **P** è la classe dei linguaggi che possono essere decisi da una macchina di Turing deterministica che impiega tempo polinomiale.

- **NP** è la classe dei linguaggi che ammettono un verificatore che impiega tempo polinomiale.

Problemi P ed NP

- **P** è la classe dei linguaggi che possono essere decisi da una macchina di Turing deterministica che impiega tempo polinomiale.

- **NP** è la classe dei linguaggi che ammettono un verificatore che impiega tempo polinomiale.
- **Equivalente:** è la classe dei linguaggi che possono essere decisi da una macchina di Turing non deterministica che impiega tempo polinomiale.

Due problemi in P . . .

Raggiungibilità in un grafo

$PATH = \{\langle G, s, t\rangle \mid G \text{ grafo che contiene un cammino da } s \text{ a } t\}$

Numeri relativamente primi

$RELPRIME = \{\langle x, y\rangle \mid 1 \text{ è il massimo comun divisore di } x \text{ e } y\}$

. . . e due problemi in NP

Problema del circuito Hamiltoniano

$HAMILTON = \{\langle G \rangle \mid G \text{ è un grafo con un circuito Hamiltoniano}\}$

Numeri composti

$$COMPOSITES = \{\langle x \rangle \mid x = pq, \text{ per gli interi } p, q > 1\}$$

P = NP ?