



# SMaLE-App: Imparare l'IA giocando

## *Laboratorio*

Bari, 20 Febbraio 2024

Roberta Raineri  
Dipartimento di Scienze Matematiche G.L. Lagrange  
Politecnico di Torino



Con il sostegno di  
Fondazione  
Compagnia  
di SanPaolo

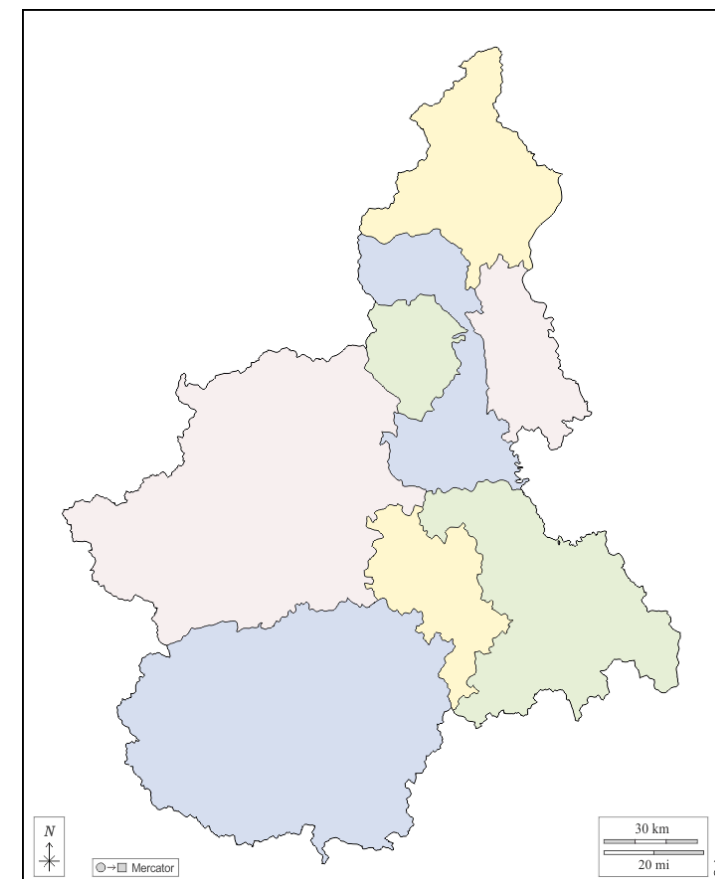


## STEP 1. Costruzione della città

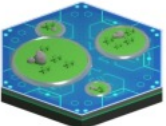


Esempio di problema di **SODDISFACIMENTO VINCOLI**

- VARIABILI
  - Chi sono? Definizione delle variabili
  - Quali valori possono assumere? Dominio
- VINCOLI
  - Quali condizioni devono soddisfare?

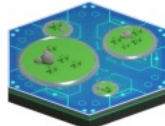


Esempio classico di problema di  
soddisfacimento vincoli  
Problema di colorazione delle mappe



Vogliamo disporre le case in modo da massimizzare il nostro punteggio

	Distanza 1	Distanza 2
	10	3
	20	10
	15	5

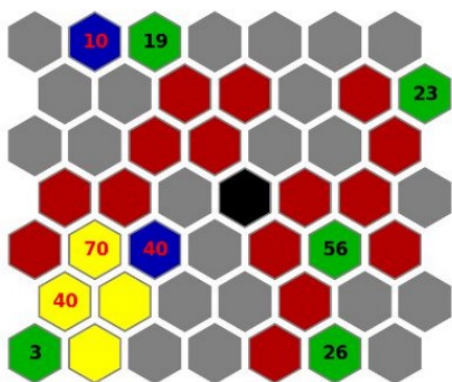
**Ma** dobbiamo rispettare dei vincoli

	Distanza 1	Distanza 2
	6	12
	8	14
	6	12

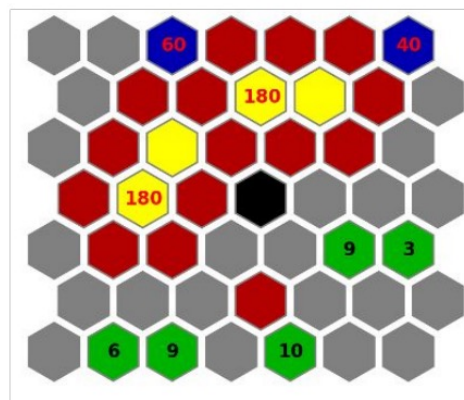


Indichiamo in giallo gli ospedali, in blu le scuole, in verde i parchi e in rosso le case.  
Vediamo un esempio di **STRATEGIA**

1) No Rules:  
Score 287



Score 497



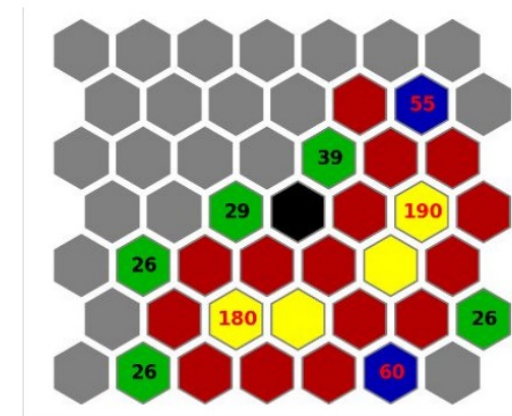
Massimizziamo il numero di case a distanza 1 da un ospedale

Score 530



Tutte le scuole devono avere 3 case a distanza 1

Score 631



Tutti i parchi devono avere almeno 2 case a distanza 1



## STEP 2. Ospedale

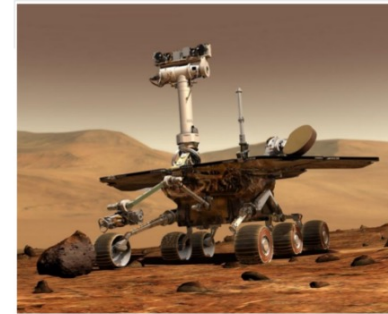
Esempio di problema di **RICERCA EURISTICA**

Supponi di volere andare dal punto A al punto B. Come scegli la strada migliore?

GPS



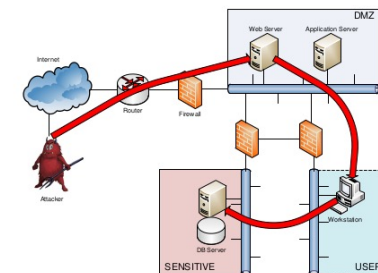
Robotics



Video Games



Network Security





Supponi di volere andare dal punto A al punto B. Come scegli la strada migliore?

- **RICERCA NON INFORMATATA**
  - Parto dal punto A ed esploro lo spazio fino a raggiungere B senza usare nessuna informazione
- **RICERCA INFORMATATA** o **RICERCA EURISTICA**
  - Utilizzi una logica che sfrutta informazioni aggiuntive per trovare più rapidamente la soluzione migliore



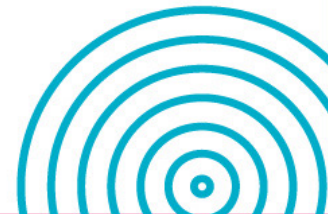
Come risolve l'intelligenza artificiale il nostro problema?

Utilizza un algoritmo di RICERCA EURISTICA.

Suddivide il problema in sotto problemi più piccoli: portare un letto per volta a destinazione

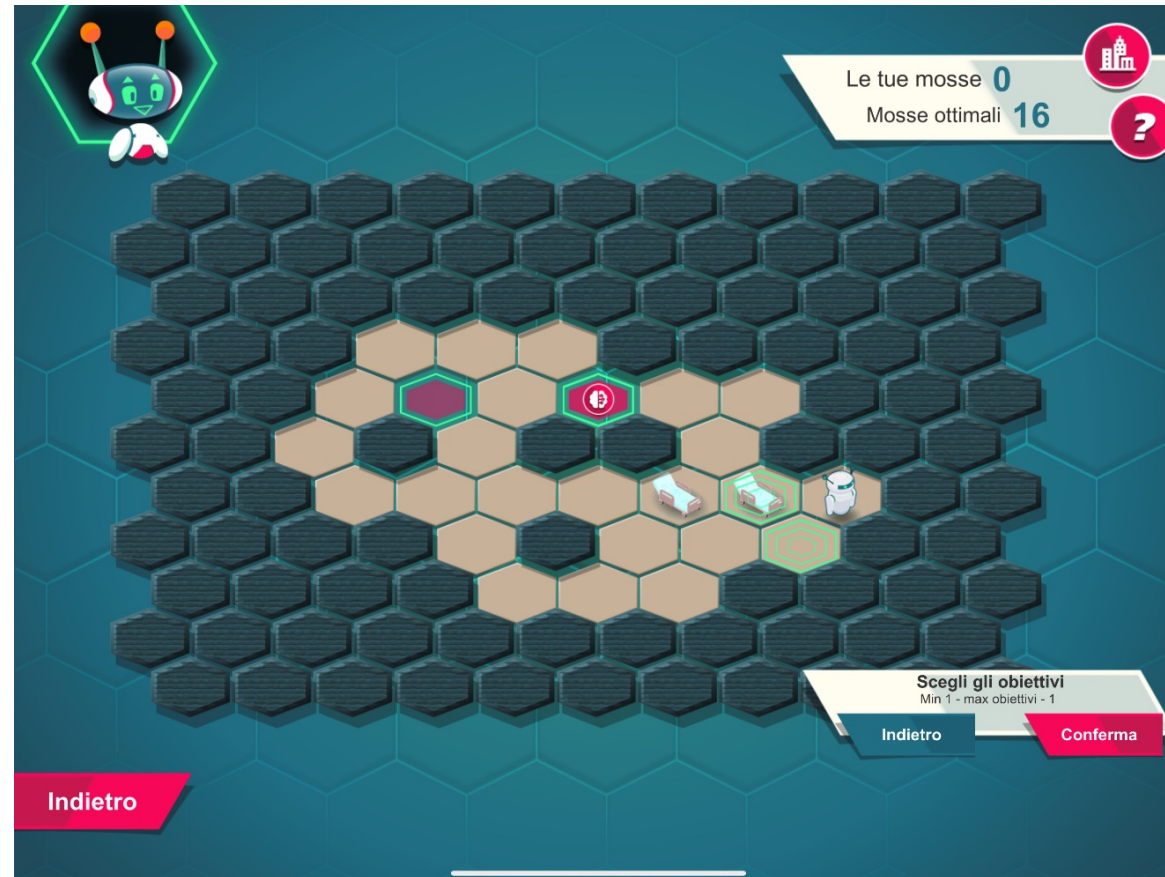


Questo metodo funziona sempre?





Questo metodo funziona sempre?





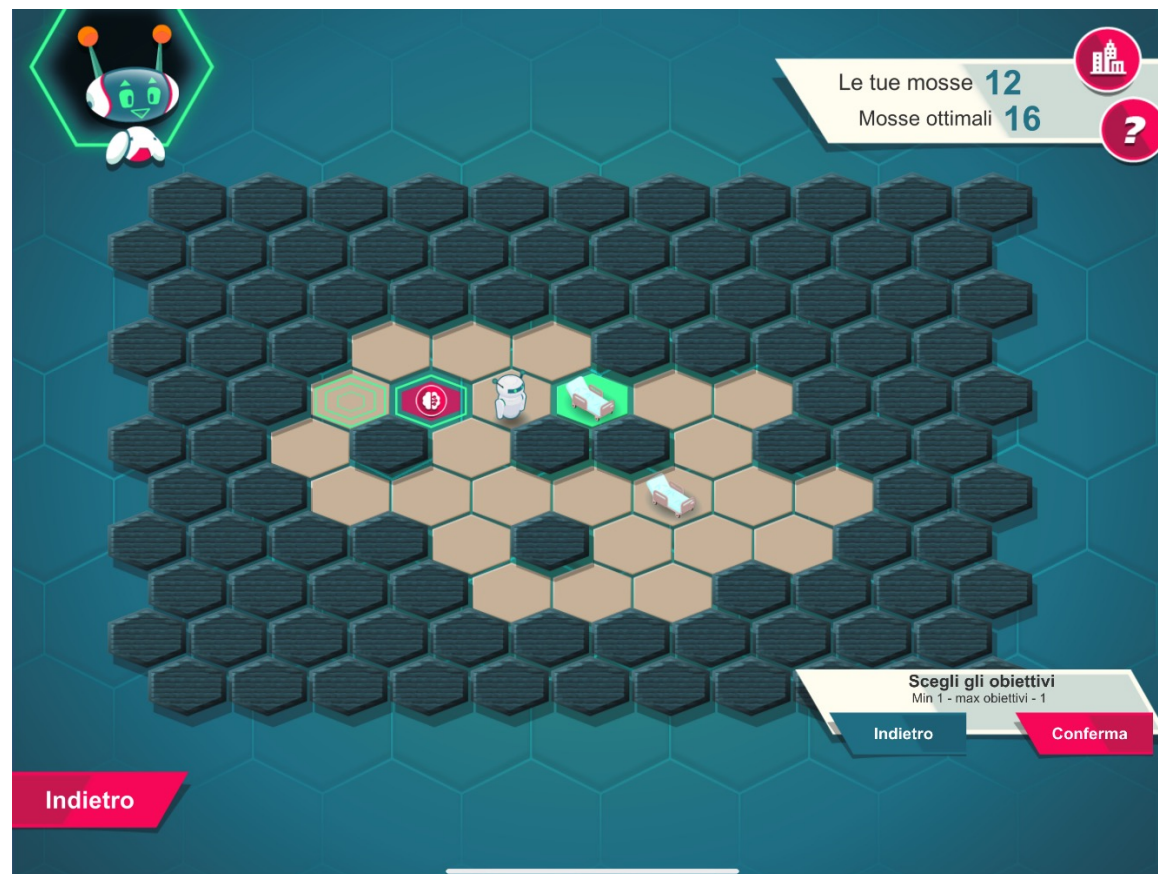
Questo metodo funziona sempre?





Questo metodo funziona sempre? **NO**

La ricerca euristica  
utilizzata cerca di risolvere  
un problema più semplice  
che non necessariamente  
coincide con quello  
generale

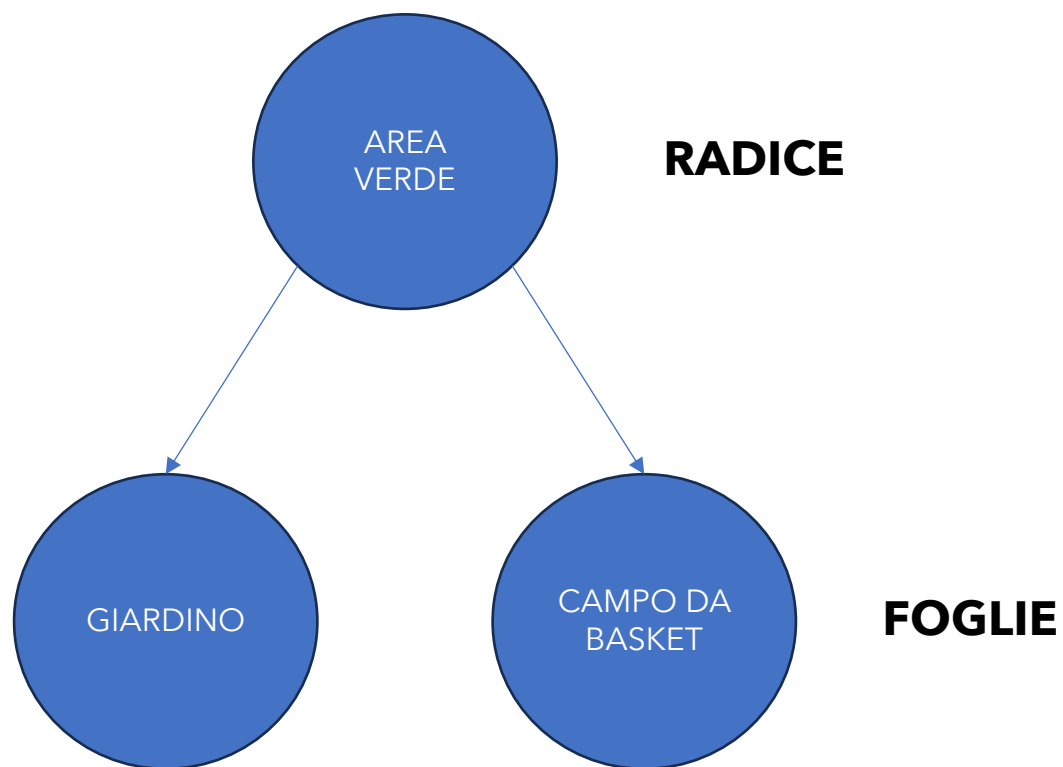




## STEP 3. Parco

Esempio di problema di **ALBERI DECISIONALI**

Rappresentiamo tutte le alternative possibili → **MODELLIZZAZIONE DEL PROBLEMA**



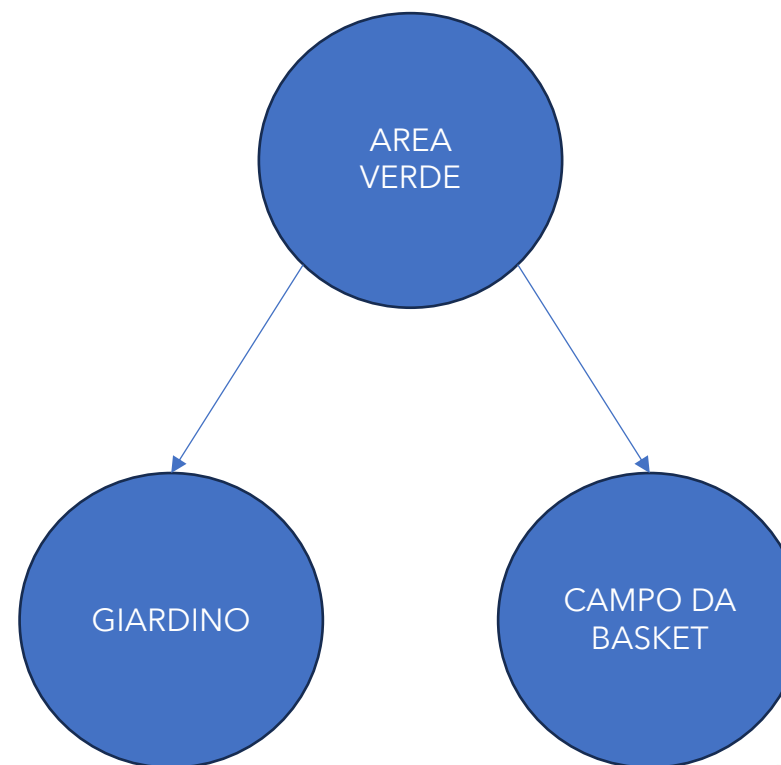
## STEP 3. Parco

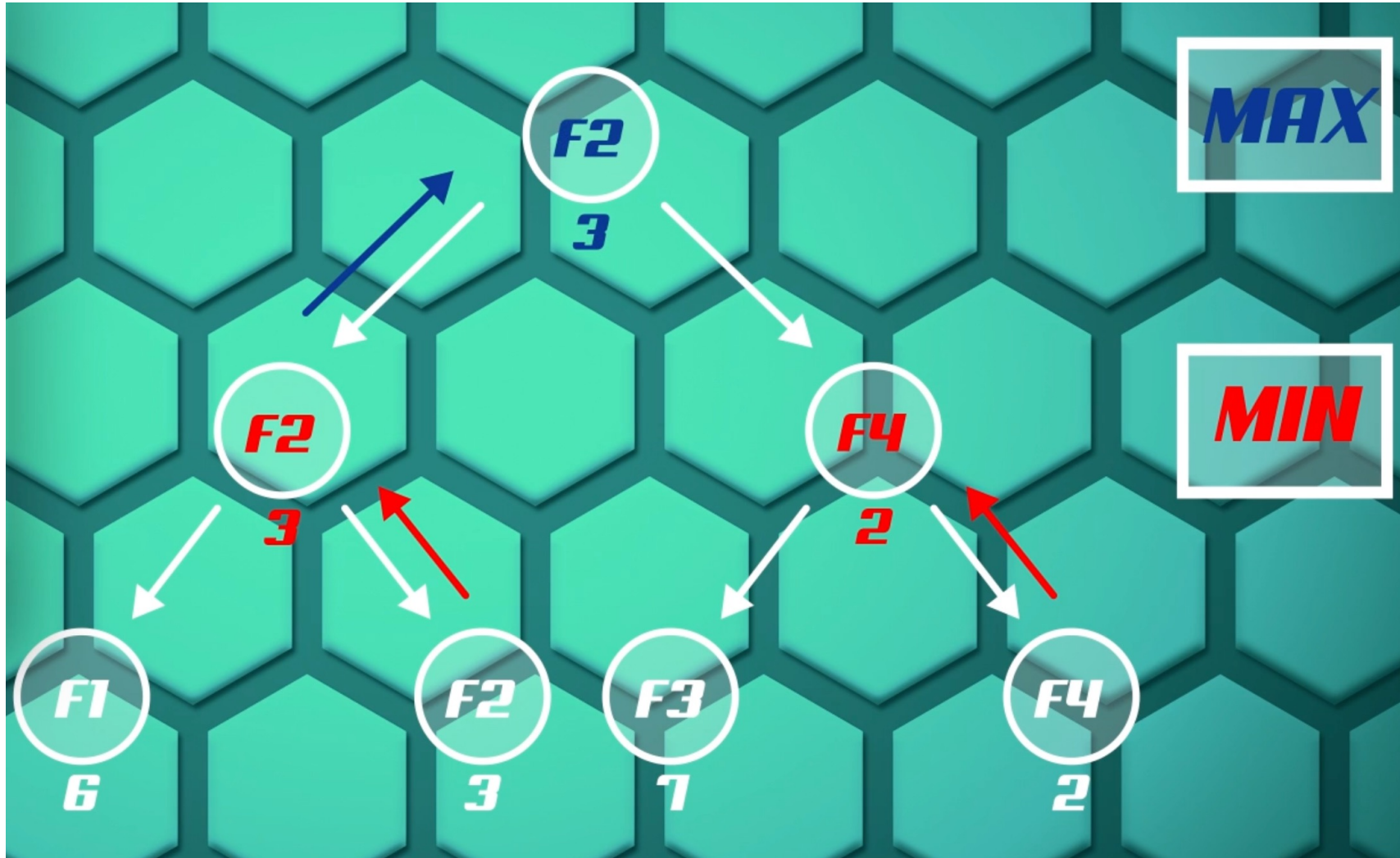
Come prendiamo le decisioni?

**Algoritmo MIN-MAX** : esempio di strategia iterativa  
Trovare la migliore tra le soluzioni peggiori

Due giocatori: MIN e MAX con obiettivi opposti.  
Noi vogliamo rendere felici i cittadini (MAX) con la strategia meno dannosa per l'ambiente (MIN)

- Parto dalle foglie
- Quando gioca MAX prendo la foglia con il punteggio più alto
- Quando gioca MIN prendo la foglia con il punteggio più basso

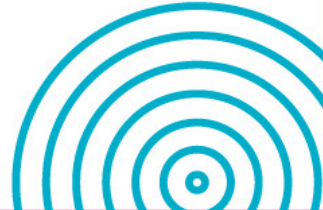




## STEP 4. Teoria dei Giochi

Il nostro parco si è trasformato ora in un problema di **TEORIA DEI GIOCHI**

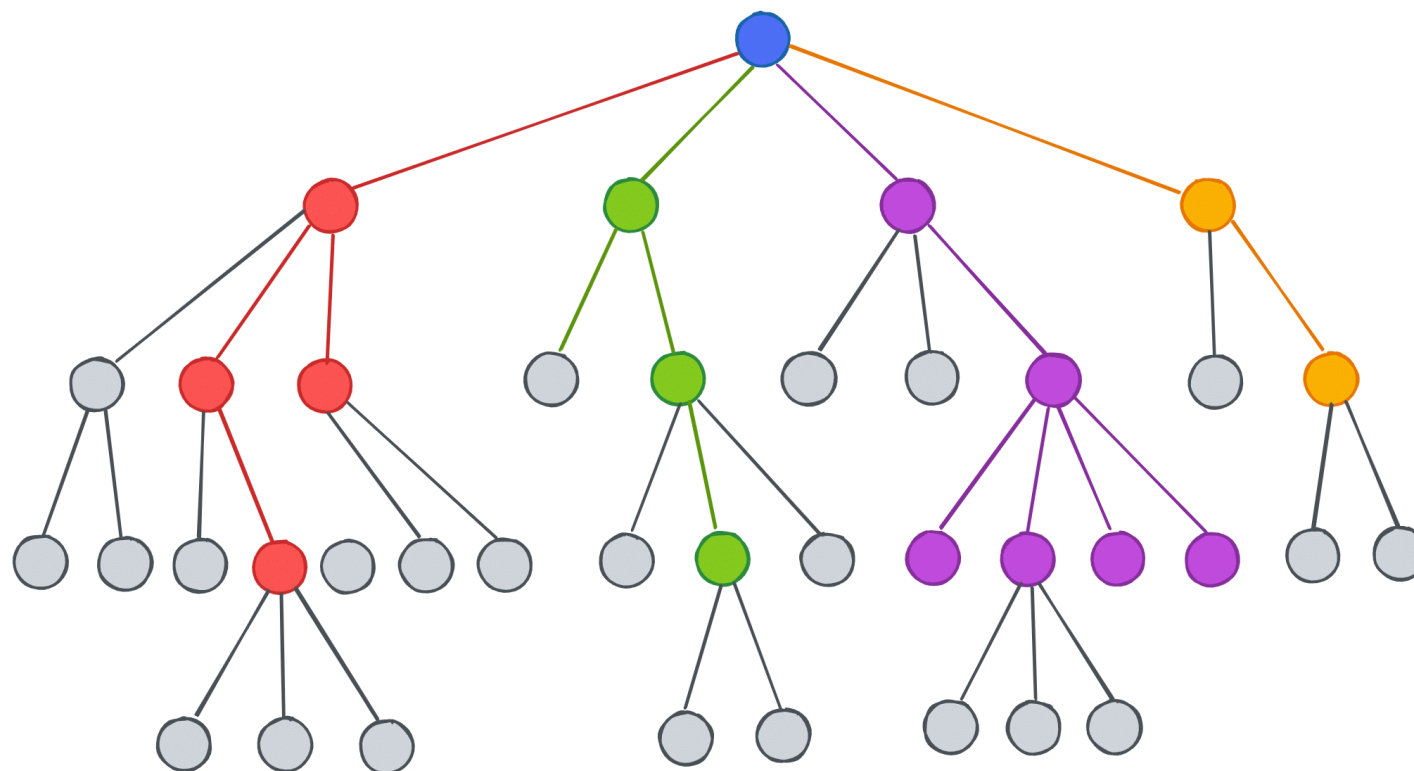
- Semina e Raccogli (→ Gioco **FORZA QUATTRO**)
- Raccogli spazzatura (→ Gioco del **NIM**)





## STEP 4.1 Forza Quattro

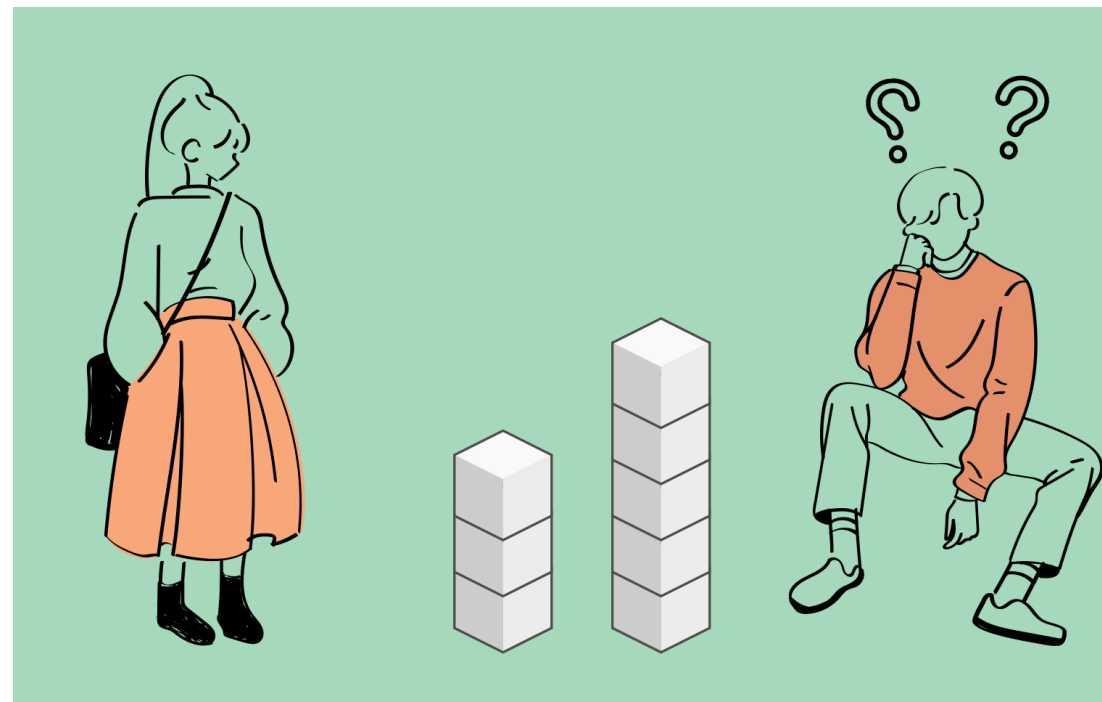
Tutte le alternative possibili vengono rappresentate tramite un ALBERO DECISIONALE.  
In base alla scelta dell'avversario valutiamo le varie alternative e vediamo quale ci porta alla vittoria.



## STEP 4.2 NIM

### Regole del gioco (a 2 colonne)

- Si gioca in 2.
- Ci sono **2** pile, una con 3 cubi e l'altra con 5.
- Si possono levare quanti cubi si vogliono ma solo da una pila.
- Chi leva l'**ultimo cubo** dell'**ultima** pila rimasta vince!

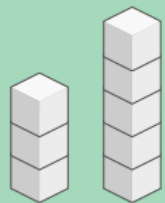


Siano date due colonne, una con  $x$  e l'altra con  $y$  oggetti, dove  $x < y$  (ad esempio 3 oggetti nella colonna 1 e 5 oggetti nella colonna 2, così  $3 < 5$ ).

La **strategia vincente** per il primo giocatore consiste nel **pareggiare le colonne**

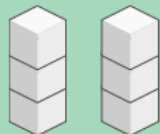


## ESEMPIO DI STRATEGIA VINCENTE



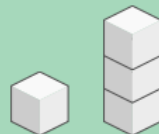
C1 C2

[Round 1],  
Giocatore A  
pareggia: leva  
2 da C2



C1 C2

[Round 2],  
Giocatore B



C1 C2

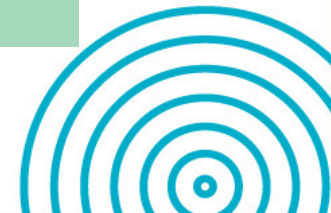
[Round 3],  
Giocatore A  
pareggia di  
nuovo: leva 2



[Round 4],  
Giocatore B

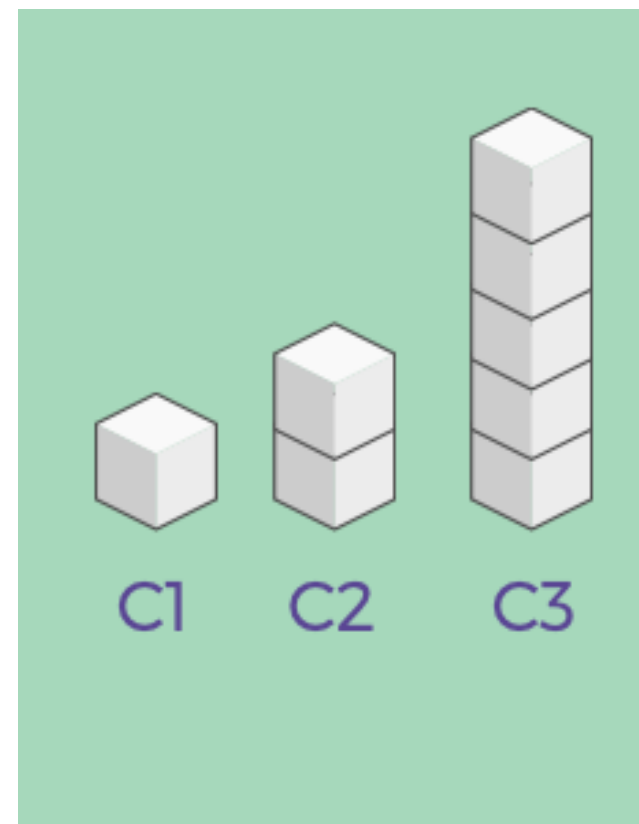


**Giocatore A  
vince!**



Il gioco si complica se ci sono **più di 2 colonne**  
Ad esempio, giochiamo il caso [1,2,5].

La strategia vincente è molto complicata ed è suggerita  
dal **teorema di Bouton**:  
"Le mosse vincenti sono quelle che mi portano nelle  
posizioni con **somma di Nim** uguale a 0".





## SOMMA DI NIM

Per prima cosa ricordiamo come si trasformano i numeri in **BINARIO** e ricordiamo che in base due  $1+1 = 10$

decimale	binario
1	001
2	010
3	011
4	100
5	101
6	110
7	111
8	1000
9	1001



## SOMMA DI NIM

Scrivo i numeri delle colonne in *binario*.

Faccio la somma dei numeri in binario ma non riporto.

### STEP 1

La colonna C1 ha un solo elemento, in binario  $\rightarrow$  001

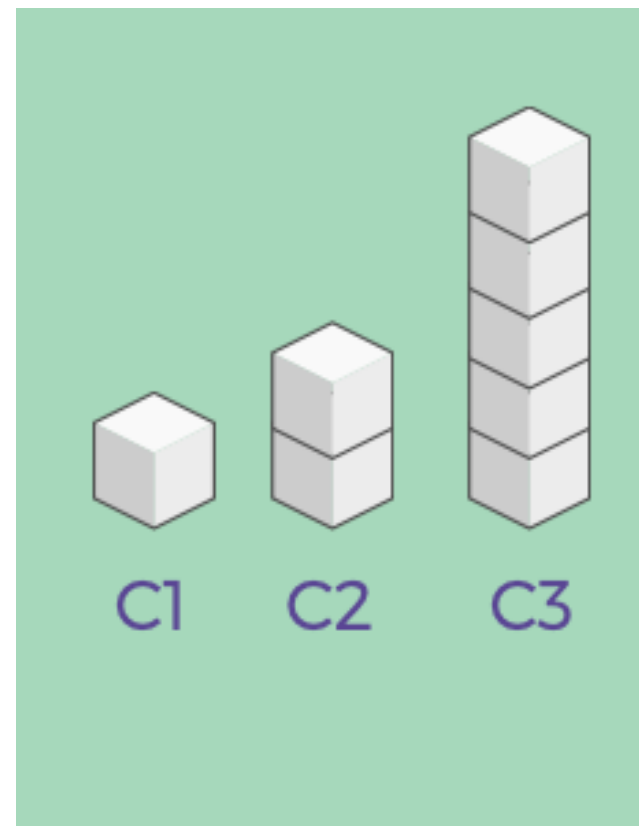
La colonna C2 ha 2 elementi, in binario  $\rightarrow$  010

La colonna C3 ha 5 elementi, in binario  $\rightarrow$  101

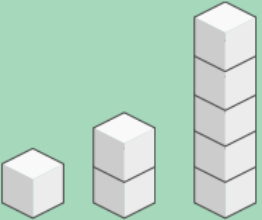
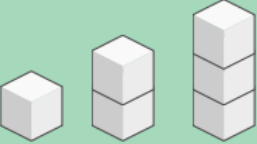
### STEP 2

Calcoliamo la somma per colonne senza riporto

$$\begin{array}{r} 001 \\ 010 \\ 101 \\ \hline 110 \end{array}$$



Levando 2 cubi da C3 ho fatto una mossa vincente perché la **somma di Nim è 0!**

 <p>C1 C2 C3</p>	<table style="border-collapse: collapse;"> <thead> <tr> <th style="border-right: 1px solid black; padding: 5px;">decimale</th> <th style="padding: 5px;">binario</th> </tr> </thead> <tbody> <tr> <td style="border-right: 1px solid black; padding: 5px;">1</td> <td style="padding: 5px;">001</td> </tr> <tr> <td style="border-right: 1px solid black; padding: 5px;">2</td> <td style="padding: 5px;">010</td> </tr> <tr> <td style="border-right: 1px solid black; padding: 5px;">5</td> <td style="padding: 5px;">101</td> </tr> <tr style="border-top: 1px solid black;"> <td style="border-right: 1px solid black; padding: 5px;">Nim 1+2+5</td> <td style="padding: 5px;">110</td> </tr> </tbody> </table>	decimale	binario	1	001	2	010	5	101	Nim 1+2+5	110	 <p>C1 C2 C3</p>	<table style="border-collapse: collapse;"> <thead> <tr> <th style="border-right: 1px solid black; padding: 5px;">decimale</th> <th style="padding: 5px;">binario</th> </tr> </thead> <tbody> <tr> <td style="border-right: 1px solid black; padding: 5px;">1</td> <td style="padding: 5px;">001</td> </tr> <tr> <td style="border-right: 1px solid black; padding: 5px;">2</td> <td style="padding: 5px;">010</td> </tr> <tr> <td style="border-right: 1px solid black; padding: 5px;">3</td> <td style="padding: 5px;">011</td> </tr> <tr style="border-top: 1px solid black;"> <td style="border-right: 1px solid black; padding: 5px;">Nim 1+2+3</td> <td style="padding: 5px;">000</td> </tr> </tbody> </table>	decimale	binario	1	001	2	010	3	011	Nim 1+2+3	000
decimale	binario																						
1	001																						
2	010																						
5	101																						
Nim 1+2+5	110																						
decimale	binario																						
1	001																						
2	010																						
3	011																						
Nim 1+2+3	000																						

Una configurazione a somma di Nim 0 si dice **SICURA**.

Da una configurazione insicura si può sempre tornare ad una configurazione sicura, mentre non è vero il viceversa.

La **STRATEGIA VINCENTE** è lasciare sempre l'avversario in una configurazione sicura.





City

**TEST**

Sustainability



684 / 666

Inhab



**GRAZIE  
PER L'ATTENZIONE**

*You have unlocked a new item in the city book, tap this button on the municipality to discover it now!*

CONTINUE A DIVERTIRVI  
**GIOCANDO** PER SCOPRIRE  
COSA SI NASCONDE DIETRO  
**L'INTELLIGENZA ARTIFICIALE**

ES (2)

