

Soluzioni secondo progetto ASD a.a. 2019/2020

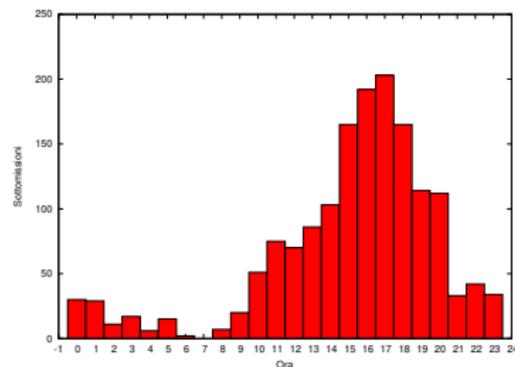
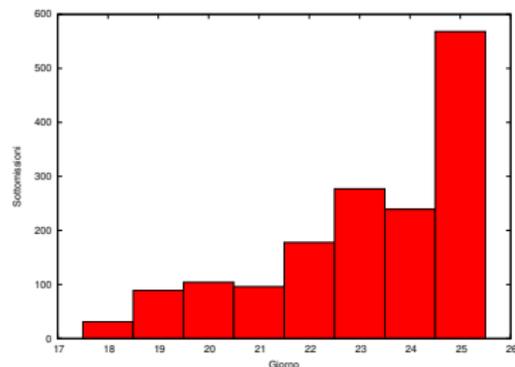
STAR WARS

IL RISVEGLIO DELL'ALGORITMO
(LA GARA DEGLI SGUSCI)

Marta Fornasier, Antonio Bucchiarone, Cristian Consonni

27 maggio 2020

Numero sottoposizioni: 1582



- ▶ 82 gruppi partecipanti, di cui 55 gruppi hanno fatto almeno una sottoposizione;
- ▶ 188 studenti iscritti, di cui 142 appartenenti a gruppi che hanno fatto almeno una sottoposizione;
- ▶ 12 ore di ricevimento;

Risultati

Punteggi (classifica completa sul sito)

- ▶ $P < 30$ → progetto non passato
- ▶ $30 \leq P < 40$ → 1 punto bonus (16 gruppi, 38 studenti)
- ▶ $40 \leq P < 60$ → 2 punti bonus (16 gruppi, 42 studenti)
- ▶ $P \geq 60$ → 3 punti bonus (21 gruppi, 58 studenti)
- ▶ $P \geq 70$ e presentazione della propria soluzione
→ +0.5 punti bonus (7 gruppi, 18 studenti)

https://judge.science.unitn.it/slides/asd19/classifica_prog2.pdf

Il problema

Masyu

Il progetto è ispirato ad un problema conosciuto come Masyu ¹. Masyu è un puzzle della casa editrice Nikoli ed è un problema NP-COMLETE.

- ▶ Non è quindi possibile individuare la soluzione ottima in tempo polinomiale (se $P \neq NP$).

¹<https://en.wikipedia.org/wiki/Masyu>

Casi rettangolari

Se è noto che il percorso che risolve il problema è un rettangolo, trovare la soluzione è semplice. Per farlo è sufficiente individuare le coordinate dei quattro vertici e collegarli.

⇒ Punteggio: 30/100.

Creare un percorso valido

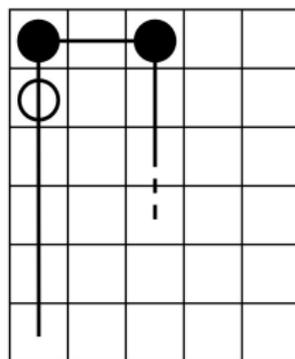
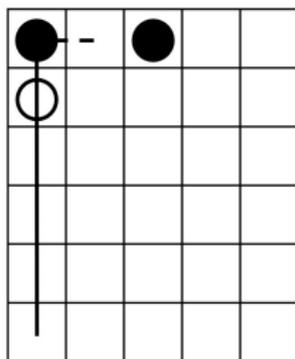
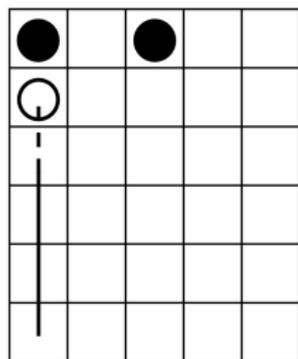
Per la creazione di un percorso valido, ad esempio, possono essere utilizzate delle funzioni come le seguenti:

- ▶ `findBlack`: una funzione che cerca se muovendosi di due passi in una certa direzione si ha la possibilità di trovare un anello nero.
- ▶ `findWhite`: una funzione che cerca se muovendosi di un passo in una certa direzione si ha la possibilità di trovare un anello bianco.

Creare un percorso valido

- ▶ `fromBlackGoOn`: una funzione che, partendo da un anello nero a cui si è arrivati da una precisa direzione, completa le regole richieste.
- ▶ `fromWhiteGoOn`: una funzione che, partendo da un anello bianco a cui si è arrivati con da precisa direzione, completa le regole richieste. Con un ulteriore parametro, tiene traccia del fatto se sia necessario o meno svoltare, dopo aver attraversato in modo rettilineo l'anello.

Esempio - Creare un percorso valido



In questo esempio:

- ▶ viene trovato un anello bianco (con `findWhite`);
- ▶ cercando di completare le regole (con `fromWhiteGoOn`) viene trovato un anello nero;
- ▶ anche per questo anello nero, si cerca di soddisfare le regole (richiamando `fromBlackGoOn`);
- ▶ facendo ciò, viene incontrato un ulteriore anello nero (e viene nuovamente chiamata la funzione `fromBlackGoOn`).

Chiusura del percorso

Il ritorno al punto di partenza può essere fatto usando una **visita dfs**. Per ottenere un risultato migliore, durante la visita si possono esplorare prima direzioni che avvicinano il punto corrente alla destinazione.

Esempio:

Per andare dal punto $(0, 0)$ al punto $(N - 1, M - 1)$, mi muoverò preferibilmente verso D o verso R .

Casi con pochi anelli - Backtrack

Se gli anelli sono pochi, possiamo usare backtrack. Ad esempio, ogni volta che incontriamo un anello nero e dobbiamo svoltare, consideriamo ed esploriamo entrambe le direzioni ortogonali a quella con cui siamo arrivati.

⇒ Punteggio (circa): 45/100.

Nota: come **non** utilizzare backtrack

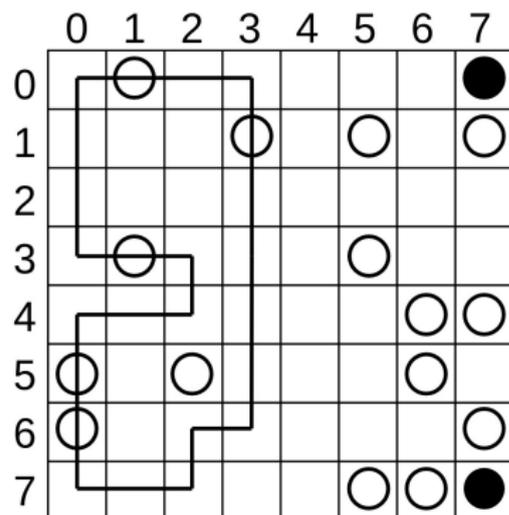
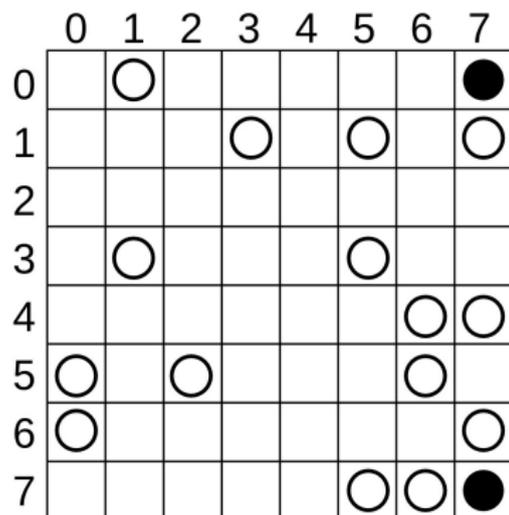
Una soluzione backtrack che in ogni punto (non solo quando incontra gli anelli) esplora tutte le possibili direzioni, risulta troppo costosa (quindi troppo lenta).

Caso generico - Ricerca locale

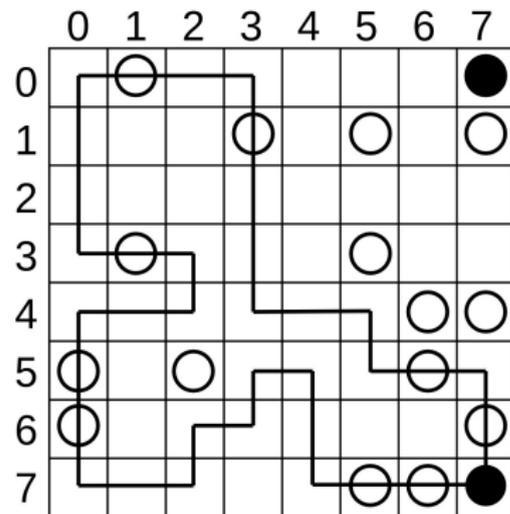
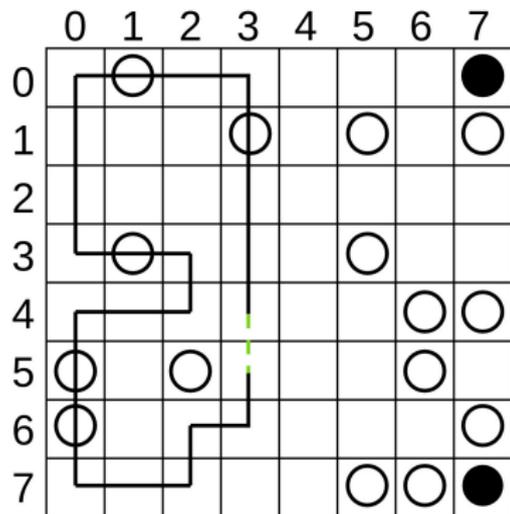
- ▶ Si crea un percorso valido (di una dimensione ridotta). Ci si muove sulla griglia cercando di prendere più anelli possibili (e se non si trova un anello nelle vicinanze, ci si muove in modo casuale).
 - ▶ **Ricerca locale:** si cerca di ingrandire il percorso scollegando due punti connessi, provando ad avanzare partendo da uno dei due e cercando poi di ricongiungersi all'altro.
 - ▶ Questa idea può essere applicata anche partendo da un percorso base costruito seguendo un preciso "pattern".
 - ▶ Ad esempio, andando dall'angolo in alto a sinistra a quello in basso a destra (e tornando indietro).
- ⇒ Punteggio (circa): 65/100.

Nota: i punti da scollegare devono essere scelti in modo da non violare le regole.

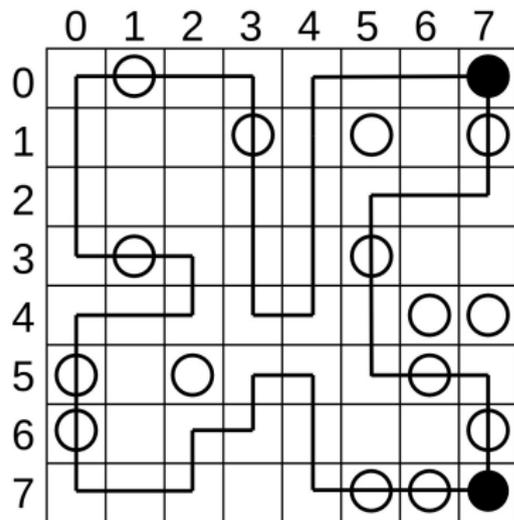
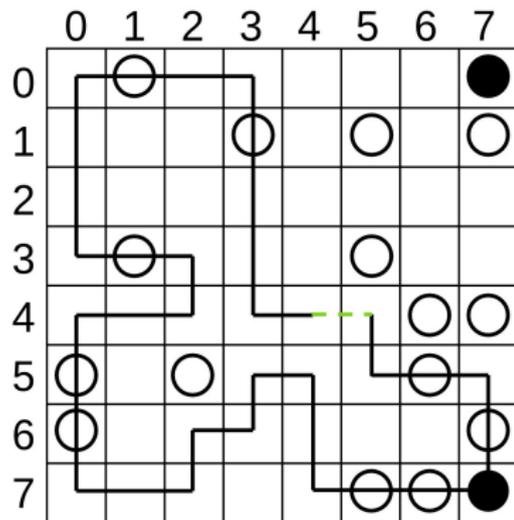
Esempio - Ricerca locale (I)



Esempio - Ricerca locale (II)



Esempio - Ricerca locale (III)



...Che la Forza sia con voi!

