

Progetto di Matematica Computazionale

Laurea magistrale in Informatica

Chess Tactics For Begginers

Wolfram Knights

A. Accornero, A. Bianchi, M. Frega, G. Magazzù

Anno Accademico 2022/2023



Indice

1	Introduzione	3
2	Scelte Progettuali	5
2.1	Package Chess Package by arneeide	5
2.2	Dataset	6
2.3	PGN o FEN	6

3	Funzionalità	8
3.1	Messaggio scelta nome utente	8
3.2	Giocare le partite	8
3.3	Personalizzazione Scacchiera	11
4	Conclusioni	13

1 Introduzione

Il seguente package di *Wolfram Mathematica*, si rivolge a utenti principianti nel mondo degli scacchi con lo scopo di allenare e migliorare l'abilità nel riconoscere uno scacco matto in una mossa. Le partite che l'utente potrà giocare, sia con le pedine nere sia con le pedine bianche, vengono estratte casualmente da uno specifico dataset. Per l'utilizzo del package non sono richieste competenze specifiche, né per quanto riguarda l'utilizzo di *Wolfram Mathematica* né per quello che concerne il mondo degli scacchi. L'unico minimo requisito richiesto è quello di conoscere le regole di base e il funzionamento del gioco degli scacchi.

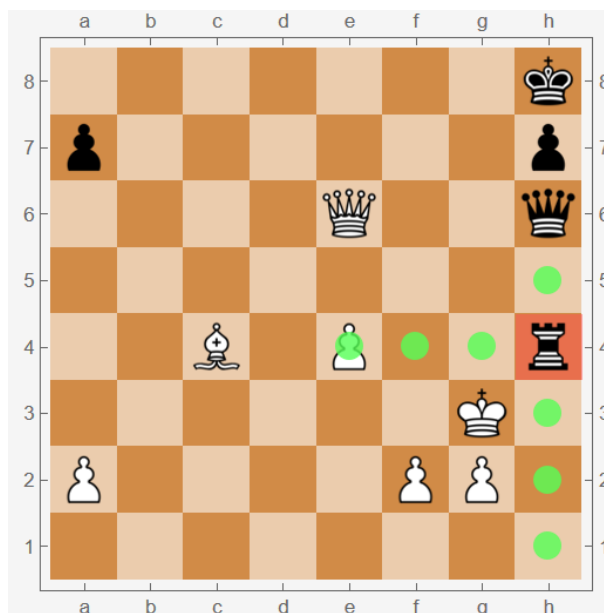


Figura 1: Esempio di partita estratta dal dataset. In questo caso l'utente muove le pedine nere ed ha selezionato la torre in posizione $h4$. I pallini in verde mostrano le mosse possibili per quella pedina.

Lo sviluppo di un progetto di questo tipo è stato motivato dal fatto che nella community di *Wolfram Mathematica*, nonostante ci fossero diversi progetti sul gioco degli scacchi, nessuno di essi si concentrava specificatamente nella **risoluzione di problemi scacchistici**, e più in particolare nell'individuazione di uno **scacco matto in una mossa**.

L'altra motivazione principale che ha spinto il gruppo a realizzare tutto ciò è data dall'interesse personale del gioco degli scacchi da parte dei membri: infatti, alcuni di essi sono dei principianti, mentre altri giocatori più esperti. L'utilizzo di un package di questo tipo potrebbe risultare utile ad entrambi per poter migliorare.

In questa documentazione discutiamo di come è stato sviluppato il progetto, quali sono state le scelte progettuali intraprese, quali le principali funzionalità implementate e alcune conclusioni finali.

2 Scelte Progettuali

2.1 Package Chess Package by arneeide

Abbiamo fin da subito deciso di usufruire di un package sviluppato in *Wolfram Mathematica*, così da avere già implementate le funzionalità di base del gioco, come per esempio mostrare graficamente la scacchiera, la posizione delle pedine, le mosse disponibili, etc. Tra i vari pacchetti a disposizione abbiamo optato per quello che più rappresentava le nostre esigenze di sviluppo e che possedeva una buona documentazione da cui attingere in caso di necessità, per questo la soluzione migliore ci è sembrata: *Chess Package by arneeide* (da ora in poi Chess Package). Il pacchetto contiene al suo interno i seguenti file:

- **Chess.pdf** : Un manuale in cui viene spiegato come installare il pacchetto, le opzioni possibili da applicare alla scacchiera e come leggere i file PGN.
- **Chess.wl** : File contenente il codice necessario per il corretto funzionamento del package.
- **Chess_Kasparov.nb**: File contenente del codice per caricare un dataset di PGN e mostrare le partite sulla scacchiera.
- **Chess_package.nb** : Notebook di Mathematica in cui viene mostrato (a livello implementativo) come muovere le pedine, la lista delle mosse effettuate, le opzioni disponibili da poter applicare alla scacchiera e come visualizzare le partite, caricate tramite formato PGN, sulla scacchiera.
- **pieceImages.mx** : File contenente le immagini delle pedine bianche e nere.

2.2 Dataset

I dati a nostra disposizione, sono ricavati da un Dataset di [Kaggle](#), contenente più di 11.000 partite realmente disputate. Lavorando con il dataset in questione ci siamo resi conto che alcune partite risultavano essere 'indentate male', ovvero rappresentate secondo una struttura erronea, e il pacchetto *Chess-Package*, utilizzato per caricarle, non riusciva a riconoscere le partite stesse, restituendo così degli errori. Per risolvere tale problema abbiamo passato i dati in formato *.txt* ad un programma sviluppato da noi in Java per risolvere le problematiche riscontrate. Una volta fatto ciò abbiamo deciso di visualizzare sulla scacchiera la partita fino alla penultima mossa, non facendo vedere all'utente la mossa corretta da effettuare. In questo modo le partite che vengono mostrate nella scacchiera sono tutte rappresentati una posizione in cui individuare uno scacco matto in una mossa.

2.3 PGN o FEN

Il **PGN** (Portable Game Notation) è un formato di facile lettura che registra le mosse di una partita (in notazione algebrica standard) e qualsiasi dato correlato come i nomi dei giocatori, il vincitore/sconfitto, e la data in cui è stata giocata la partita.

Il formato **FEN** (Forsyth-Edwards Notation) è una notazione abbreviata per descrivere una posizione statica di una scacchiera. È composta da sei parti separate da spazi. Queste parti rappresentano la disposizione dei pezzi sulla scacchiera, il turno di gioco, i diritti di arrocco, la possibilità di cattura "en passant", il conteggio delle mosse senza catture o spostamenti di pedone e il numero di mossa corrente. La stringa FEN viene utilizzata per comunicare e registrare posizioni di scacchi in modo conciso e standardizzato.

```

[Event "Panchev"]
[Site "?"]
[Date "1946.??.??"]
[Round "5"]
[White "Marjanovic, Slavoljub"]
[Black "Zhivanov"]
[Result "0-1"]
[Opening "Si Closed, 2...Nc6"]
[ECO "B23"]
[WhiteElo "2420"]
[Comment "Smother Checkmate"]

```

```

1.e4 c5 2.Nc3 Nc6 3.Nge2 g6 4.d3 Bg7
5.Be3 Nd4 6.g3 Nf3# 0-1

```

Figura 2: Partita rappresentata in formato PGN

Esempio: La sequenza *"rnbqkbnr/pppppppp/8/8/8/8/PPPPPPPP/RNBQKBNR"* indica la posizione iniziale della scacchiera

Nel nostro caso abbiamo deciso di procedere con una rappresentazione di tipo PGN, in modo tale da avere un quadro generale sulla partita, e non la posizione statica della tastiera, anche perché il pacchetto *Chess Package* carica le partite utilizzando questo formato.

3 Funzionalità

3.1 Messaggio scelta nome utente

Una delle prime funzionalità sviluppate è stata quella di permettere all'utente la scelta del proprio nickname che viene visualizzato mentre sta giocando.

All'avvio viene mostrato un messaggio pop-up in cui si richiede l'inserimento obbligatorio del proprio nome, ma non ci sono vincoli sulla lunghezza né sull'utilizzo di caratteri speciali. Il nome scelto non potrà essere una stringa vuota ed eventuali spazi tra una parola e l'altra verranno soppressi.

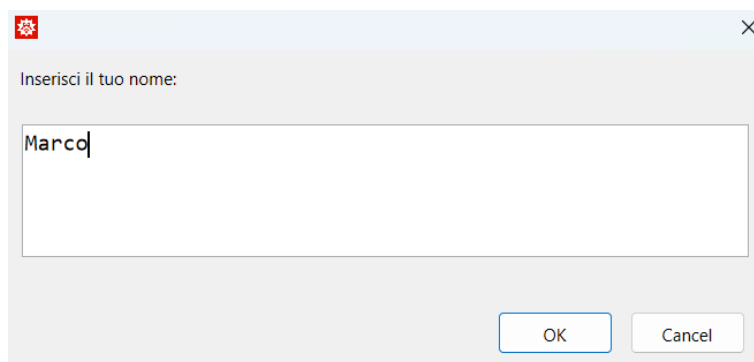


Figura 3: Messaggio pop-up scelta nome

3.2 Giocare le partite

Una volta scelto il proprio nome utente, il giocatore si troverà davanti una rappresentazione grafica della scacchiera con le rispettive pedine bianche e nere in posizione iniziale. In questa fase l'utente non sarà in grado di interagire con la scacchiera in quanto disabilitata, poichè essa rappresenterebbe la schermata iniziale del gioco.

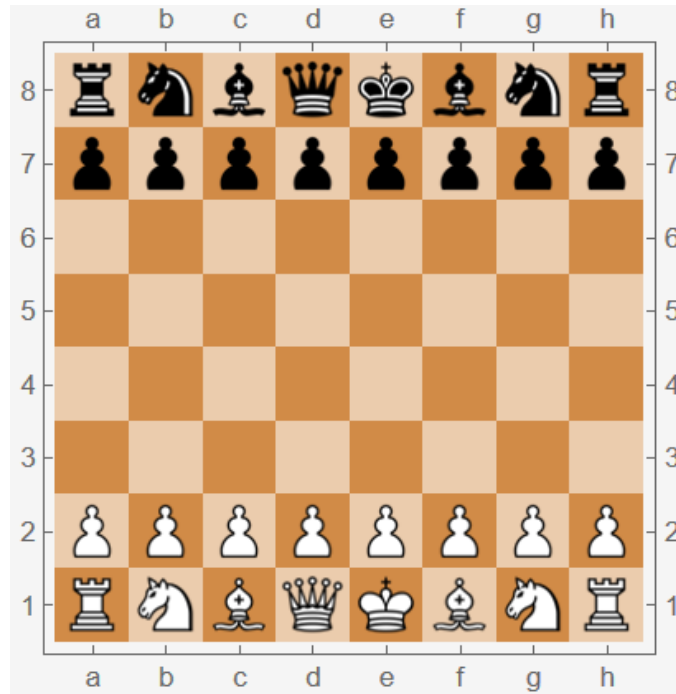


Figura 4: Scacchiera nella schermata principale

Al di sotto della scacchiera sarà presente una struttura tabellare dove in alcune celle saranno presenti dei bottoni con il quale l'utente potrà interagire, mentre nelle altre sono presenti delle immagini e dei messaggi di output.

Il primo bottone che vediamo è quello di **Nuova Scacchiera**, cliccandolo l'utente genererà una nuova partita (viene estratta casualmente dal dataset) dove verranno mostrate le pedine in una posizione specifica e l'utente dovrà essere in grado di capire quale pezzo muovere, a seconda se spetta la mossa al bianco o al nero, per trovare lo scacco matto. Per implementare questa funzionalità, generiamo un numero casuale che ha una lunghezza pari a quante sono le partite contenute nel dataset, estraiamo la partita in questione e visualizziamo la posizione della partita fino all'ultima mossa, senza ovvero

quella che l'utente dovrà indovinare. Per distinguere quando l'utente deve giocare con i bianchi o con i neri, facciamo riferimento a 3 caratteri presenti alla fine di ogni partita del dataset: **1-0** indica la vittoria del bianco, **0-1** indica la vittoria del nero. Di conseguenza la risoluzione del problema consisterà nel muovere una pedina del giocatore vincente.

È presente anche il bottone **Restart**, cliccabile solo dopo aver premuto il pulsante *Verifica Mossa*, il quale riporta la scacchiera alla schermata principale, così da permettere all'utente se giocare una nuova partita o quella appena giocata. Per implementarla abbiamo semplicemente passato il PGN alla scacchiera così che mostrasse la schermata dell'inizio con tutte le pedine in posizione iniziale.

Il bottone **Rigioca Partita** serve per rigiocare nuovamente la partita appena disputata dall'utente e si abilita solo quando l'utente ha giocato almeno un match e ha cliccato il pulsante **Restart**. Per implementare la funzione è bastato memorizzare la partita appena giocata dall'utente e ricaricarla nella scacchiera nel caso in cui l'utente premesse il bottone.

Il bottone **Mostra soluzione** stampa la soluzione corretta per lo scacco matto: per implementarlo abbiamo ricavato l'ultima mossa dal dataset e l'abbiamo mostrata in output.

Il pulsante **Verifica Mossa** che stampa un messaggio di errore in caso di mossa corretta o un messaggio di vittoria in caso contrario. L'implementazione di questa funzionalità avviene nel seguente modo: viene confrontata l'ultima mossa della partita con la mossa appena effettuata dal giocatore. Se le due corrispondono allora l'utente ha fatto la mossa corretta, perciò viene stampato il messaggio di vittoria, in caso contrario il messaggio di errore.

I pulsanti sono dinamici e vengono abilitati e disabilitati ogni qualvolta è necessario limitare o garantire la possibilità di effettuare determinate opera-







	Marco sta giocando	
Nuova Scacchiera	Restart	Verifica Mossa
Mostra Soluzione		Rigioca Partita
Dimensione Scacchiera	Colora Scacchiera:	Reset Colore
		

Figura 5: Tabella dei pulsanti

zioni: in questo modo è possibile far seguire all'utente un processo guidato che gli permetta di limitare eventuali errori.

3.3 Personalizzazione Scacchiera

Per quanto riguarda la personalizzazione della scacchiera sono presenti **3** pulsanti:

- **Dimensione Scacchiera:** Permette il ridimensionamento della scacchiera tramite 4 dimensioni di default, passando dalla dimensione più piccola a quella più grande in ordine crescente. L'implementazione è stata immediata perché, grazie al package *Chess-Package*, bastava passare un numero ad un parametro della scacchiera per ridimensionarla.
- **Colora Scacchiera:** Colora la scacchiera secondo il colore scelto tramite il ColorSetter RGB. Anche in questo caso l'implementazione è stata facilitata da **Chess-Package** in quanto era necessario passare un solo parametro contenente il colore RGB alla scacchiera.
- **Reset Colore:** Ristabilisce il colore di default della scacchiera allo stato iniziale. L'implementazione è identica chiaramente al bottone *Colora Scacchiera*.

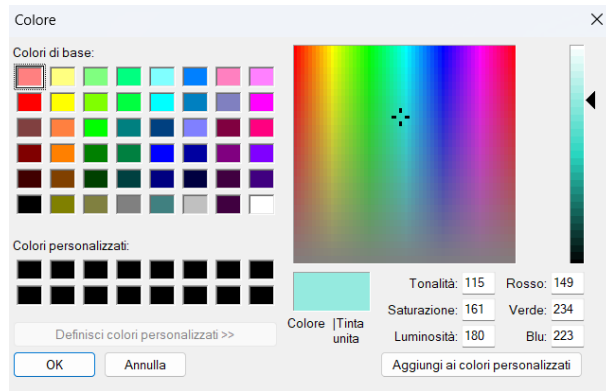


Figura 6: Color Setter per scegliere il colore RGB con cui colorare la scacchiera

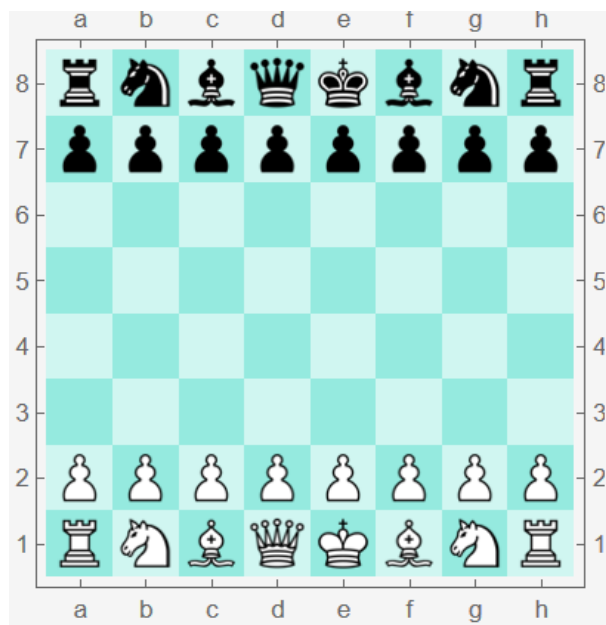


Figura 7: Esempio di scacchiera colorata

4 Conclusioni

Con il seguente documento sono state mostrate le principali scelte implementative e progettuali effettuate per la realizzazione del progetto. Durante la realizzazione del pacchetto sono state riscontrate alcune difficoltà di percorso: in primis non era chiaro sin da subito come utilizzare il pacchetto *Chess-Package* e come implementarlo nel nostro progetto per farlo funzionare; successivamente la lettura e caricatura delle partite in formato PGN non è stata di facile intuizione, riscontrando diverse problematiche nel far giocare esclusivamente l'ultima mossa all'utente e non l'intera partita. Nonostante le difficoltà il risultato ottenuto è per noi più che soddisfacente: chiaramente il pacchetto realizzato può essere espanso e migliorato per garantirne una maggior fruizione, rendendo il gioco più competitivo e stimolante da giocare.