

Ferretti

## Contents

# 1 Obbiettivi del progetto

Lo scopo di questo progetto è implementare il gioco degli scacchi in una versione CLI<sup>1</sup>. Auspicabilmente, vorremo fare in modo che l'utente possa giocare in due modi:

1. Una versione locale ove il programma gestisce solo la logica del gioco e il risultato su schermo.
2. Una versione in cui si gioca contro un modello locale di CPU.

## Punti cardini dello sviluppo

1. È un progetto didattico. Esistono centinaia di progetti già funzionanti online.
2. Vorremmo che fosse efficiente e non solo funzionante.
3. Auspichiamo una buona lettura del progetto da parte di terzi. Per questa ragione siamo interessati a documentare bene il progetto e le scelte progettuali prese.

# 2 Lista delle cose da fare

Presentiamo una lista di cose da fare per ogni componente del progetto. Nota: L'ordine conta! Trovi prima le cose più necessarie e in fondo quelle meno.

## 2.1 Board

1. Controllare se effettivamente il costruttore di Board funziona.
2. Aggiungere controlli ed eccezioni per la costruttore della posizione a parte dal fen.
3. Fare funzioni più piccole a partire da funzione da fen to board.

## 2.2 Engine

1. Aggiungere dei test per tutto.
2. Aggiungere stampa risultati e menu di continuazione.
3. Prendere il numero come string e poi parsarlo a uint8\_t. Attualmente fa crashare il programma se in input si mette un -n.
4. Sistemare *isMate*.
5. Sistemare *saveGame*.

---

<sup>1</sup>Command Line Interface

6. Sistemare *loadGame*.
7. Sistemare *takePlayerTurn*.
8. Sistemare *takeEngineTurn*.
9. Sistemare *playGameVsEngine*.
10. Aggiungi supporto multipli salvataggi

### **2.3 Coords**

1. Aggiungere dei test per tutto.

### **2.4 Printer**

1. Aggiungere dei test per tutto.
2. Aggiungere Menu per scelta colore.

### **2.5 Piece**

1. Aggiungere dei test per tutto.

### **2.6 Runner**

1. Modificare il file .bat
2. Creare il file .bash

### **2.7 Nuove funzionalità**

1. Implementare una versione di gioco con orologio funzionante in tempo reale.
2. Creare un motore per giocare contro il computer offline.

### **2.8 Idee per Evaluation**

#### **2.8.1 Generale:**

- delta di materiale
- mobilita' pezzi
- centralita' pezzi
- king's safety
- pawn islands

- pedoni doppi/tripli
- pedoni in prossimità di promozione (7o e 2o rank)
- pinned pieces
- numero mosse legali totali
- occupazione delle ali.
- triangolazione re per guadagnare tempo.
- dono greco.
- sacrificio pezzi minori per vantaggio sviluppo.
- forchetta
- alfieri in fianchetto
- attacchi di scoperta
- pezzi sospesi (+malus)
- pezzi difesi (+bonus)
- sacrificio semplificativo
- evitare il cambio nello svantaggio
- cambio forzante
- valore nel far perdere l'arrocco
- cavallo messo in avamposto non scacciabile
- scacco di scoperta

### **2.8.2 Endgame (+middlegame)**

- mobilità re
- pedoni passati
- pedoni passati supportati da altri pedoni
- Cattura pedoni centrali contro laterali.
- Mosse Zugzwang

## 2.9 Idee per la search

### 2.9.1 Generale

- cercare prima e piu' a fondo se si da' scacco
- cercare prima ogni mossa che cattura un pezzo
- cercare piu' a fondo se si vince un pezzo
- cercare per ultimo ogni mossa che perde materiale
- cercare prima ogni mossa che centralizza un pezzo (o che si avvicina al re opposto)

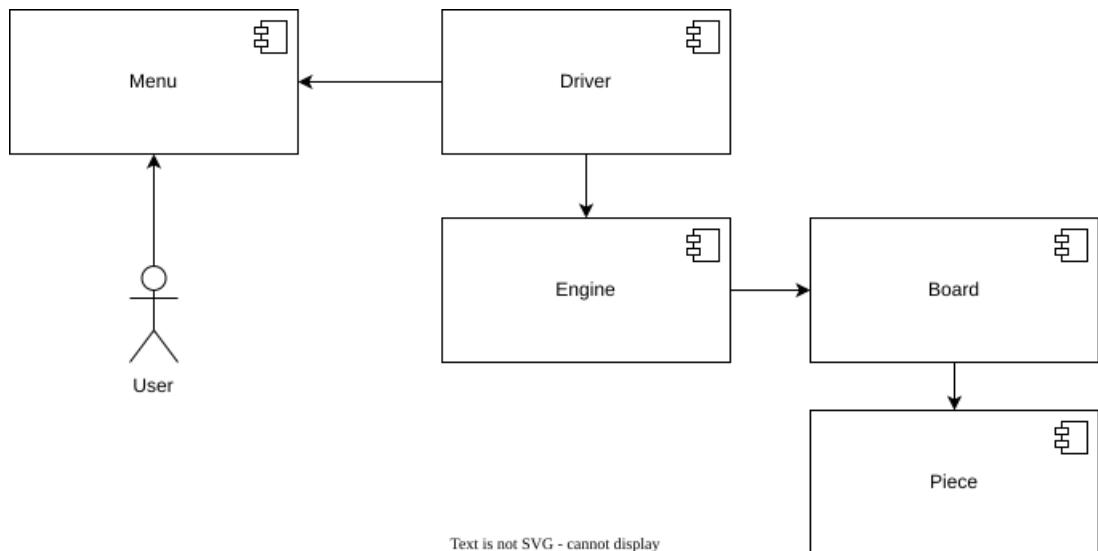
### 2.9.2 Middlegame

- Cercare in profondità di attaccare le caselle deboli dell'avversario

### 2.9.3 Endgame

- cercare prima le mosse che diminuiscono la mobilità del re avversario
- cercare prima le mosse che obbligano il re avversario ad avvicinarsi al bordo scacchiera
- cercare prima le promozioni di pedoni

## 3 Architettura del progetto



### 3.1 Componenti generali

**Coords** È una classe che rappresenta le coordinate nella scacchiera. Quindi, l'insieme delle righe sarà così composto:  $R := \{A, B, C, D, E, F, G, H\}$

Mentre l'insieme delle colonne:  $C := \{1, 2, 3, 4, 5, 6, 7, 8\}$

Una coordinata non è altro che:  $RXC$ .

### 3.2 Board

**Descrizione classe:** Rappresenta la foto di una scacchiera in questo momento.

**Costruttori** Sono presenti due costruttori:

1. Costruttore vuoto: Genera la posizione classica degli scacchi.
2. (std::string): Costruttore che prende un FEN costruendo la relativa posizione.

bitmap board

La bitmap contiene in totale 3 informazioni:

1. Indice, che viene trasformato in coordinate
2. Tipo del pezzo tramite 3 bit
3. Colore del pezzo tramite 1 bit



La posizione  $board[0]$  rappresenta la casella: a8. Procedendo con l'indice di  $board$ , iniziando dalla casella a8 (casa in alto a sinistra nella scacchiera), si procede da sinistra verso destra e dall'alto verso il basso.

Ripostate alcune corrispondenze per spiegare ulteriormente:

- Indice 7: h8
- Indice 9: b7
- Indice 18: c6
- Indice 37: f4

**Metodi:**

### 3.3 Engine

**Descrizione classe:** E' la classe che gestisce il gioco. Media l'interazione tra l'utente e le componenti.

**Costruttori** E' presente un solo costruttore:

1. Costruttore vuoto: serve a inizializzare le componenti interne.

```
Board board;
```

**Metodi:**