

### Overview

The program is design as a chess game using C#. It allows players to take turn making moves using standard algebraic notation and validate those moves according to chess piece location , side and the rule of chess.

### Components

1. ChessGame Class - this class represents the chess game itself. It contains methods for initialising the game, displaying the board , making moves and checking for game over conditions
2. Board Class - The Board class represents the chessboard . It initialises the board with pieces at the beginning of the game , displays the board, and handles moving pieces.
3. ChessPiece Class - this class represents a chess piece. It defines the types (eg. bishop,knight) and side (white or black) of each piece.
4. PieceLogic Class: This class contains methods for validating moves for each type of chess piece. It checks whether a move is valid based on the rules of chess

### Functionality

- Upon starting , the chessboard is set up with pieces placed in their starting positions
- Players take turn making moves by specifying the source and destination squares using prompt example
- Moves are validated using PieceLogic class , which ensure that each move complies with the rules of chess for respective piece types.
- The game continues until a checkmate or a stalemate condition is met , signalling the end of the game.

### How it works

1. ChessGame class build the game by creating a new instance of the Board class and setting up the initial board configuration
2. Player take turns making moves by specifying the source and destination
3. The program will validate each move using the PieceLogic class to ensure it adheres to the rules of chess.
4. If a move is valid , the piece is moved on the board and the turn switches to the other player.
5. The game continues until a checkmate or a stalemate condition is detected, at which point the game ends.

### Summary

The program provides a basic implementation of a chess game in C# , simulating the usage of encapsulation , abstraction , inheritance , polymorphism and composition to create and proceed the game of chess. Encapsulation through Board Class , Abstraction through allowing players to interact with the game through MakeMove and Display. Inheritance through ChessPiece class. Polymorphism through isValidMove validation and composition through managing game states via delegating certain task relations.