CHESS Project report

Name: Fahad Ansar
Student number: 6203384

How to RUN AND OPERATE my program:

Main.py is the main file that you need to run (running instructions are given in README.txt)

-→PLEASE *TURN PUTTY INTO FULL SCREEN MODE (or large enough so the board doesn't cuts off)*

First thing that you will see is a set of some rule (really basic rules) like
- type ‘q’ when asked about exiting the game after both players moves
- index range is from 1-8 .
- after turn of each player the board would be ROTATED. due to which, the player whose turn it is will be at bottom part of the board always.

Then you will be given an option to play with a mode out of below,

0 - Human vs Human Mode
1 - AI vs Human Mode
2- Human vs Human (Testing Mode)

you will be asked to enter a number before the name of the modes to get into that mode. If you don't press one of above numbers or press wrong key you will get into human vs human mode.

After Entering into..

Human vs Human Mode:

- You will be asked for the name of the player A and player B
- Then board will be print with your side of the board at bottom (always)
- And you would be asked for indexes of the piece you want to move (please enter between 1 – 8)
- then you would be asked for indexes of the unit/box where your want to move it (please enter between 1 – 8)
- Then your piece would be moved with considering all the rules of chess
- After this board would be rotated (with a message of board rotation) for other persons turn and he/she would be on bottom side of the board
- And this goes on until someone won or its a tie

AI vs Human Mode:

- You will be asked for the name of the player A and player B would be AI
- Then board will be print with your side at of the board bottom (always)
- And you would be asked for indexes of the piece you want to move (please enter between 1 – 8)
- then you would be asked for indexes of the unit/box where your want to move it (please enter between 1 – 8)
- Then your piece would be moved with considering all the rules of chess
- It would AI's turn now, it will take some seconds to calculate moves and then it will perform its move
- and then game comes back to your turn
- This will keep happening until you or AI won or its a tie

Human vs Human (**Testing Mode with your own created initial state)**:

- You will be asked for the name of the player A and player B
- Then you would be given options to place the pieces on the board by you choice and then play into human vs human mode.
- It will go through all pieces that are initially present on the board and ask you whether you want it on the board or not. If you don't want it simply press any key ('Enter' key recommended) and if you want it then type 'Y' and press enter and it will take indexes of where you want to place it.
- Then board will be printed with your side of the board at bottom (always)
- And you would be asked for indexes of the piece you want to move (please enter between 1 – 8)
- then you would be asked for indexes of the unit/box where your want to move it (please enter between 1 – 8)
- Then your piece would be moved with considering all the rules of chess
- After this board would be rotated (with a message of board rotation) for other persons turn and he/she would be on bottom side of the board
- And this goes on until someone won or it a tie

How does my chess system WORKS:

My chess system has three modes to play in

1. Human vs Human:

Human versus Human, human is placed against a human. First names of the player are taken as input to display it while printing board, win state, etc. Then current board state is printed with indexes and names of the sides on chess board.
Program takes input of the indexes of the piece you want to move and indexes you want that pieces to move to. program takes care of wrong inputs like string instead of indexes, etc.
Once a player have performed it move then its turn of the other guy. The board is flipped and checked for Checkmate, Win state, pawn promotion and Tie State. Now other player is asked same information to perform a move and then checked for Checkmate, Win state, pawn promotion and Tie State as first player. This mode works on simple idea that both players making their moves turn by turn and then program checks for all the end/termination conditions of the game.

2. AI vs Human:

This mode is the based of whole program. This mode is what whole program is based on. Half of this mode works similar to human versus human mode. First it asks for the names of the players are taken as input to display it while printing board, win state, etc. Then current board state is printed with indexes and names of the sides on chess board. In this case, player 2 would be named AI.
Program takes input of the indexes of the piece you want to move and indexes you want that pieces to move to. program takes care of wrong inputs like string instead of indexes, etc.
Once human player have performed it checks for all the end states/termination states like checkmate, wind state, pawn promotion. it AI's turn. AI takes whole board as a parameter and then passes it on to minimax function that has alpha beta pruning. Which creates general tree of all combination till a certain depth. We decrease the nodes expanded by alpha beta pruning. At leaf nodes we evaluate using evaluation function (heuristic function).
I have really less heuristics because I added some heuristics but my program was taking more than two minutes at the depth of four, So I removed some costly heuristics to visit tree 5 layers deep in less with time less than 30secs.
After AI has calculated and made its mode, Human player is again given turn to move a piece and then All the checks are performed on the board (checkmate, wind state, pawn promotion, etc). This loop keeps repeating until one of the checks become true. (checkmate, wind state, etc)

Minimax function (with alpha beta prunning) psuedocode,

Minimax(parameter: depth, board, minimaxTurn, teamA, teamB, alpha, beta):

if depth is equal to zero:
evaluate the board using heuristics function
else if board Is in tie state: return MAX NEGATIVE INT
else if board is in win state: return MAX POSITIVE INT

Flip the minimaxTurn and initialize the score according to the turn
(-99999 if turn is 0 (MIN) or 999999 if turn is 1 (MAX))

Find all the legal moves and store it in some data structure

for loop on all the valid moves

copy current board to a new board
update the new board with a valid move

score = Min call with alpha beta pruning
beta = min(score , beta)
if alpha>=beta: break

score = Max call with alpha beta pruning
alpha = max(score, alpha)
if alpha>=beta: break

selecting best board from 1 layer deep

return score

Heuristics used (All Heuristics are made by me)

- Each piece in AI adds points to the heuristic sum and each piece of following kind in opponents teams decreases the score by same amount
  - Bishop – 3 points
  - Knight – 5 points
  - Rook – 50 points
  - Queen – 15 points
  - King – 5 points
- If AI queen is not present, deducts 2 points
- if opponent queen not present, adds 3 points
- if AI king is present, adds 50 points and if not present, deducts 9999
- if opponent King not present, add 9999 and opponent king present, deducts 30 points
- For each opponent bishop, deducts 1 point and for each AI bishop, add 1 point.

3. Human vs Human (Testing with custom initial board)

This mode is similar to normal human versus mode but the only difference is, In this mode you first place pieces according to your wish to create a initial state of the board and then play human versus human on that board.
The main purpose of creation of this mode is testing. Basically This mode is for marker to make marking easier.
So In the beginning, Program takes the names of both players in chess for future use. Then asks you one by one for the existence of each piece on the board and if you reply yes for a piece, it will ask you for the indexes where you want to place that piece. It asked for all the pieces that exist in initial state of a chess board. After it has asked for all the pieces it creates the desired board , prints and then starts the game on that board. It is played similar to human verses human mode. It places your side at the bottom of the board.
Then it takes input of the indexes of the piece you want to move and indexes you want that pieces to move to. program takes care of wrong inputs like string instead of indexes, etc.
Once a player have performed its move then its turn of the other guy. The board is flipped and checked for Checkmate, Win state , pawn promotion and Tie State. Now other player is asked same information to perform a move and then checked for Checkmate, Win state, pawn promotion and Tie State as first player. This mode works on simple idea, both players making their moves turn by turn and then program checks for all the end/termination conditions of the game.

Files and their usage in the project:

No.	Python File name	Usage
1.	Team.py	Has a team object which has two instantiations in the game.
2.	Pieces.py	Has all the chess pieces class objects with a piece maker class. Pieces maker class use piece objects to creates their instantiations.
3.	CMethods.py	Has all the required game methods like move validation method, Given board printing method, pawn promotion method.
4.	AI.py	The method handles complete Artificial intelligence part of the project. This has minimax function with heuristics evaluation function and other dependencies of minimax function.
5.	Main.py	This file connects all the files and form complete game. Uses different method from other files to construct a working game
6.	Board.py	This file has chess board and method to create and modify chess board