

# Assignment #1

Due date: 10/7

Total Score: 100 + (max. 30 bonus points)

Can be completed by individual or team with max. 5 members

---

**Objectives:** Implementing a **mini-max algorithm** with **heuristic functions**

**Problem:** Write a program to play a relatively simple chess endgame using a *mini-max algorithm*. *PlayerX* has a rook and king and *PlayerY* has only the king left. Of course, with more chess pieces, *PlayerX* is able to win this game. So *PlayerX* should try to win the game as quickly as possible avoiding any infinite loop or dead end. On the other hand, *PlayerY* should try to delay the game as long as possible or end the game in a draw.

---

## Required Activities

1. **Write a program** to play both/either *PlayerX* and/or *PlayerY*. Your program should define at least a data structure to represent a **state** (a chess board) and several functions/procedures, **move** (a search method), **play** (a driver or controller), **heuristic** (function to return a heuristic value), and **printBoard** (to print a chess board on a screen or in a file). You can define any other utility functions or procedures. The function, **play** (*n*) is a driver function that starts, ends, and controls the game by alternating the move up to *n* number of moves, where *n* is a parameter representing the max. number of moves allowed for each player in a game. When a king is under attack and there is no applicable next move, it returns “**checkmate**”. In this case, the player that returns “**checkmate**” loses the game. When a king is not under attack and there is no applicable next move except for the move that would put or leave the king under attack, it returns “**stalemate**”. In this case, the game ends in a draw. The function **play** should immediately stop the game either when “**checkmate**” or “**stalemate**” occurs even before the *n*th move or after a player makes *n* number of moves even if no player returns either “**checkmate**” or “**stalemate**”. **Note:** *The total number of moves is counted only based on all the moves made by one player (either PlayerX or PlayerY without double counting).*

Define two heuristic functions, **heuristicX** for *PlayerX* and **heuristicY** for *PlayerY*, each of which returns a numerical heuristic value indicating the usefulness of a move for a given chessboard. **printBoard** prints the state of each move during the game in a nice 8x8 board form to a file and on the screen.

Properly modularize the program clearly separating it into different modules (classes, package, or dll) for related data structures, functions, and procedures. Particularly, allow the heuristic function to be easily replaced whenever necessary. Document the program properly naming and commenting each module, class, function, or procedure so that one can easily understand the intention.

Programming languages to use: C++, Java, Python, and C#.

## 2. Test your program

Once your program is started, e.g., by entering a command **Play**, ask two questions:

- (a) Is this a test?
- (b) Enter maximum number of moves: (default is **35**)

**If the answer for the question (a) is Y (or Yes)**, test your program with the following 2 test cases, alternating each move by *PlayerX* and *PlayerY*.

**Test case1:** x.K(5,6), x.R(8,5), y.K(6,8) and **Test case2:** x.K(6,5), x.R(5,6), y.K(4,7) where x.K and x.R represents the coordinates of King and Rook for *PlayerX* and y.K represents King for *PlayerY*.

Each test case will be given as a file, “*testCase.txt*” that contains the coordinates of each piece as shown in the above examples. Your program should read this file to play the game. **printBoard** function should print the trace of running program in a nice format in a file named “*gameResult.txt*” and display on a screen **without prompting** to ask for a test case or output file name while running as shown in the following example:

```
-----
Game started...
Testcase1: x.K(6,5), x.R(5,6), y.K(4,7) or the chess board
-----
Board =>
Board =>
Board =>
...
Board

Number of moves made: 10
Game result: checkmate, stalemate, or max. # of moves reached
```

**Note:** **Board** represents an actual chessboard with chess pieces placed on it or at least the coordinates of each piece.

**If the answer for the question (a) is N (or no)**, it means the program is started to play a game with other team. See the instructions for **4. Champion game**.

- ## 3. Write a brief report in Word format including (a) team name, the name(s) and email addresses of all members; the percentage contribution to this assignment if the assignment was completed by a team. If a team cannot reach a consensus on individual contribution, include the individual’s claimed percent contribution with a brief description on specific tasks performed (b) a brief instruction on how to run the program (c) a brief description on the algorithm and heuristic function(s) implemented, and the game result for each test case, (d) provide reference(s) to the source of the program specifying the URL, author if any when some portion or all of the program was reused (or copied) from other people’s code, (e) optionally comments or lessons learned from this assignment.

**Note:** (a) ~ (c) are required and (d) ~ (e) are optional. However, (d) is required if you reused the source code.

**Warning:** Although the code reuse is allowed for this assignment, copying the code from other person or team in this class is strictly prohibited. Any one or team violating this rule will receive **ZERO** score for this assignment.

#### 4. Champion game

The tournament game will begin on **10/8**. Every team should participate in order to receive the basic points.

Prepare or modify your program so that it can play with other team's program, e.g., making one move at a time for a given board (by opponent's move). After a move, show the new board modified by the move.

##### Game rules:

- (a) The champion will be selected following the tournament rule that is similar to the one used in the World Cup game. All teams will be divided into groups and have a *round-robin tournament* (each team plays with all other teams within their group) to select one top-ranked competitor from each group. Top-ranked competitors have a *single-elimination tournament* (only the winner will progress to next round) to select the champion.
- (b) Initial coordinates for *PlayerX* will be given by the team playing *PlayerY*.
- (c) For each game, two teams (e.g., Team A and Team B) will play as both Player X and Player Y, alternatively. For example, if Team A plays Player X, then Team B plays Player Y and vice versa. The team that makes smaller number of moves when it plays as Player X wins the game.
- (d) Once a game is started, no team is allowed to modify any portion of the program including the heuristic function until the winner is determined. However, the program can be modified after a game to prepare for next game with other teams.
- (e) If there is a tie, each team runs the program for a test case prepared by opponent's team. The team that makes less number of moves for a given test case win the game. This process may be repeated until the winner is determined. Hopefully one can eventually win the game! If not, they can play again with an improved heuristic function.

#### 5. Grading

Grade will be based on the quality of the program, heuristic function(s), and written report.

Any team participated in the champion tournament will receive min. **20 basic points** (**not bonus points**) IF the program doesn't crash during a game, making a reasonable move based on a heuristic function. **Note:** Making a random move is not considered as a reasonable move.

**30 bonus points** will be given to the 1<sup>st</sup> place (champion) and **20 bonus points** will be given to the 2<sup>nd</sup> place winner.

//////////////////**How to Submit this assignment**////////////////////////////////////

**Include** (a) a report, (b) source code, and (c) an executable program into **ONE zipped file** and submit the **zipped file** to **Titanium**. I strongly recommend you to write your report in **Word format** so that I can provide my feedback directly in the report if necessary. No feedback will be given for a PDF format of report.