HW3

W3101: Programming Languages, C++: Homework 3

- **Due Oct 07.**
- Please submit your solutions in Courseworks, do not mail them to me.
- Please submit ONLY your source (.cpp) files, not the entire project.
- Please call your files < YourName >_HW_3 to help TAs identify your file quickly.

1. Make the Game class of HW2 an abstract class. Modify it as given below. It should have the following:
   I. A private variable called "numberOfPlayers" of type integer.
   II. A public constructor without any arguments that sets "numberOfPlayers" to 1.
   III. A second public constructor with an integer argument "n", which sets "numberOfPlayers" to "n".
   IV. A public pure virtual function called "bool isValid( )".
   V. A public pure virtual function called "bool isOver( )".
   VI. A public pure virtual function called "void PrintGamePosition( )".
2. Design a simple "Chess" program (for chess game) with only pawns and king as a child class of the "Game" class. That is, your program should have a white king, a black king, white pawns in the range 0 to 8 and black pawns, also in the range 0 to 8. The class should have the following:
   I. A private two-dimensional 8x8 array of characters, called "board" to represent the chess board. Each element may contain a chess piece. You can assume 'K' represents the white king, 'k' represents the black king, 'P' represents a white pawn and 'p' represents a black pawn. If a square doesn't have any piece, you can use a default character, say '0' to represent it.
   II. A public constructor that sets the "numberOfPlayers" to 2 in the parent class and sets all the squares of the board to '0', indicating all the 8x8 = 64 squares are empty.
   III. A public function called "void readDataFromFile(char *fileName)" that reads a chess position from a file and sets the "board" variable. I will post the code for it in Courseworks.
   IV. Implement the public function "void PrintGamePosition( )". It should print the chess board as 8 rows. Print a blank (nothing) if there is no piece in a square.
   V. Implement the "bool isValid( )" function as follows:
      i. There should be only one white king and one black king.
      ii. There should be a maximum of 8 white pawns and 8 black pawns.
      iii. No white pawn should be on the first row (row 0).
      iv. No black pawn should be on the last row (row 7).
      v. A king cannot attach the other king (i.e., no check by king).

      If all the above conditions are satisfied, return true. Otherwise, return false.
   VI. Implementing the "bool isOver( )" function is a bit more complicated, but not impossible. You can return "false", for this HW.

      Optional: If you are interested, you can write the code to check if a king is in check and has no other free (unattacked) square to go. If so, return true; otherwise return false.
   VII. Design the rest of the chess class -- list as many functions that you think will help in designing a full chess class. Write the details of your design and what the function is supposed to do, as comments in the function. No need to implement them.
      i. What data, if any, would you store in the class, apart from the board information?
      ii. What functions will you implement?
      iii. How will you alternate the moves between the two players?
      iv. When and how would decide when the game is over?

Ramana Isukapalli