

CPSC 2150 Project 1

Connect 4

Michael Ellis, Cooper Taylor, Ryan Chen, Adam Niemczura

Requirements Analysis

Functional Requirements:

(For BoardPosition.java)

- As a player, I need the pieces to stay in the position I originally put them, so I can track pieces. (public BoardPosition(int aRow, int aColumn))
- As a player, I want to see the row names, so I know the row position of a spot. (public int getRow())
- As a player, I want to see the column names, so I know the column position of a spot. (public int getColumn())
- As a player, I need to know when two positions on the board are equal, so I can compare positions and know where I can place my token. (public boolean equals(Object obj))
- As a player, I need to see a positions' row and column, so I can easily identify a position of a spot on the board. (public String toString())

(For GameBoard.java)

- As a player, I need the game board to be initialized with empty spaces so that I can start the game on a blank board. (public GameBoard())
- As a player, I need to check if a column is available for placing my token so that I can make a valid move in the game. (public boolean checkIfFree(int c))
- As a player, I need to be able to drop my token in a specific column and have the token drop to the lowest available row of that column so that I can make a move. (public void dropToken(char p, int c))
- As a player, I want to know if the last move I made resulted in a win so that I can see if I have won the game. (public boolean checkForWin(int c))
- As a player, I want the game to check if the board is full and no more moves can be made so that I know if the game ends in a tie. (public boolean checkTie())
- As a player, I need to check if I have 5 tokens in a row horizontally after making a move so that I can see if I have won the game with a horizontal line. (public boolean checkHorizWin(BoardPosition pos, char p))

- As a player, I need to check if I have 5 tokens in a row vertically after making a move so that I can see if I have won the game with a vertical line. (public boolean checkVertWin(BoardPosition pos, char p))
- As a player, I need to check if I have 5 tokens in a row diagonally after making a move so that I can determine if I've won the game with a diagonal line. (public boolean checkDiagWin(BoardPosition pos, char p))
- As a player, I want to know what is at a specific position on the game board (whether it is empty) so that I can keep track of the game state. (public char whatsAtPos(BoardPosition pos))
- As a player, I need to check if I am at a certain position on the board so that I can verify my moves. (public boolean isPlayerAtPos(BoardPosition pos, char player))
- As a player, I need the game board to provide a string representation of its current state so that I can visualize the entire game board during gameplay. (public String toString())

(For GameScreen.java)

- As a player, I need to be able to specify the column that I want to place my token in, so that I can play against my opponent how I want. (public void getInput())
- As a player, I need to know whose turn it is, so that I can play on the correct turn. (Public void printGameState())
- As a player, I need to see that I am placing a token in an invalid place, so that I can choose a valid column. (public void printGameState())
- As a player, I need to see that the game has ended, so I know when to stop playing. (public void checkGameEnd())
- As a player, I need to see who has won, so I know the result of the game. (public void checkGameEnd())
- As a player, I can input Y or N at the end of the game, so that I can start a new game. (public void checkGameEnd())
- As a player, I need to be able to see the printed out board, so that I can know what turn the opponent has made and the current state of the gameboard. (public String printBoard(String boardString))

Non-Functional Requirements

- The board should have 9 rows and 7 columns
- There should be 2 players, called X and O
- A player must have 5 tokens in a row either horizontally, vertically, or diagonally to win
- Gravity should affect the tokens
- The number of tokens in a column should not exceed the number of rows

- An error message should be returned if a player tries to place a token in an invalid column.
- The program should have 2 end conditions: win or tie
- The game will start with player X
- Players should alternate placing tokens
- The game should be able to repeat once the game has ended
- Empty positions on the board should be indicated by a single space character

System Design – (UML diagrams)

