Spalding Latham

Requirements Analysis

Functional Requirements

- Functional Requirement: User enters the row, column, and number to win numbers for the game

- Role: User

- Action: Enter the number of rows 3-25, enter the number of columns 3-25, enter the number to win > 3

-Benefit: Allows user to customize board

- User Story: As a user, I can customize the game to be how the size I want, which is really nice

- Functional Requirement: User enters what column they want their token in

- Role: User

- Action: Enter column number 0-6

- Benefit: Allows them to play the game

- User Story: As a user, I can enter the column I want my token in to try and win the game

- Functional Requirement: Program prints out the game board

- Role: User

- Action: View game board

- Benefit: Allows you to see what the previous moves are

- User Story: As a user I can view the game board to see previous moves and where I want to go next

- Functional Requirement: Program checks if there has been a win or a tie

- Role: User

- Action: See if there is a win/tie or keep playing

- Benefit: Allows you to know whether the game is over and if so who won

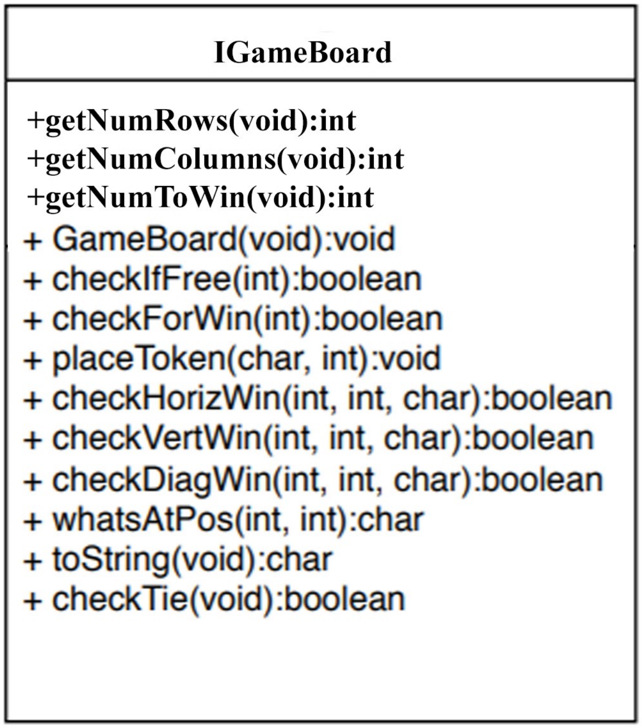
- User Story: The program outputs if there is a win or a tie, but if it doesn't I know to keep playing

Non-Functional Requirements

- System must be coded in Java
- System must run on Unix
- System must work for all possible inputs/situations of Connect 4

Deployment

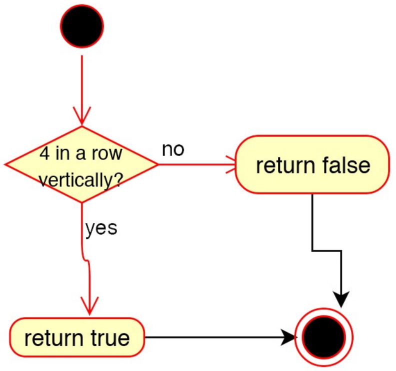
Download the provided "hw2.zip" file, and unzip it in your Unix system. Move into the hw2 directory where the Makefile is located, and from there you can type "make clean" in the terminal to remove unnecessary files, "make" to compile the program, and "make run" to run the program.



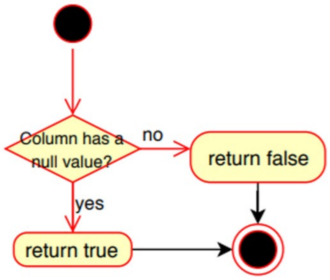
IGameBoard

+getNumRows(void):int
+getNumColumns(void):int
+getNumToWin(void):int
+ GameBoard(void):void
+ checkIfFree(int):boolean
+ checkForWin(int):boolean
+ placeToken(char, int):void
+ checkHorizWin(int, int, char):boolean
+ checkVertWin(int, int, char):boolean
+ checkDiagWin(int, int, char):boolean
+ whatsAtPos(int, int):char
+ toString(void):char
+ checkTie(void):boolean

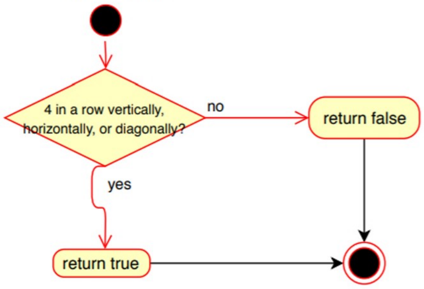
checkVertWin



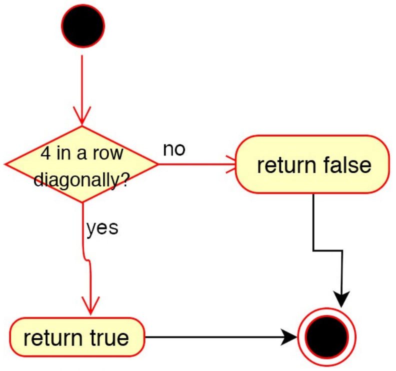
checkIfFree



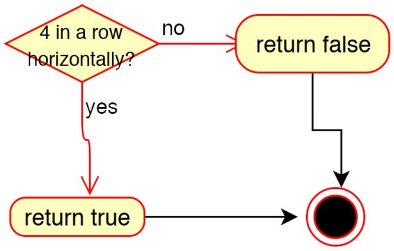
checkForWin



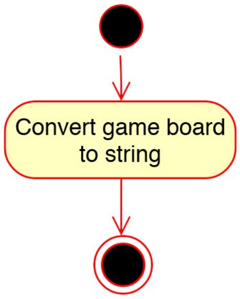
checkDiagWin



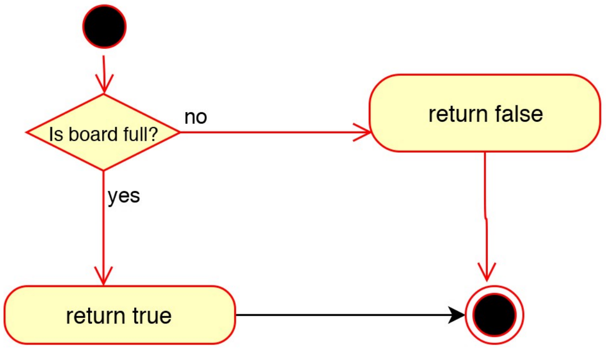
checkHorizWin



toString



checkTie

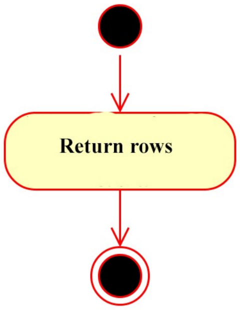


GameBoard

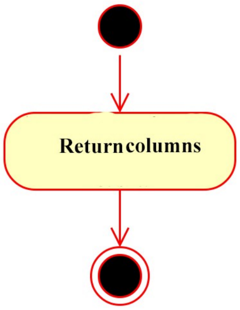
- Rows:int[1]
- Columns:int[1]
- **Num:int[1]**
- gameBoard:char[][][1]

+ GameBoard(void):void
+ **getNumRows(void):int**
+ **getNumColumns(void):int**
+ **getNumToWin(void):int**
+ placeToken(char, int):void
+ whatsAtPos(int, int):char

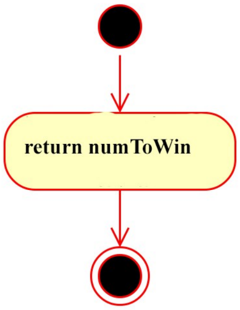
getNumRows



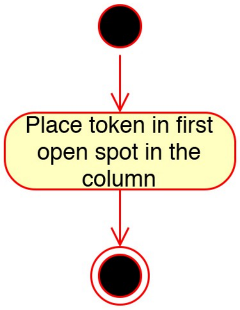
getNumColumns



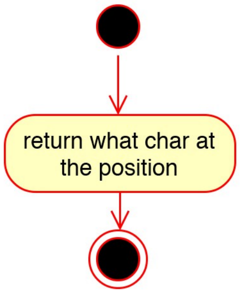
getNumToWin



placeToken

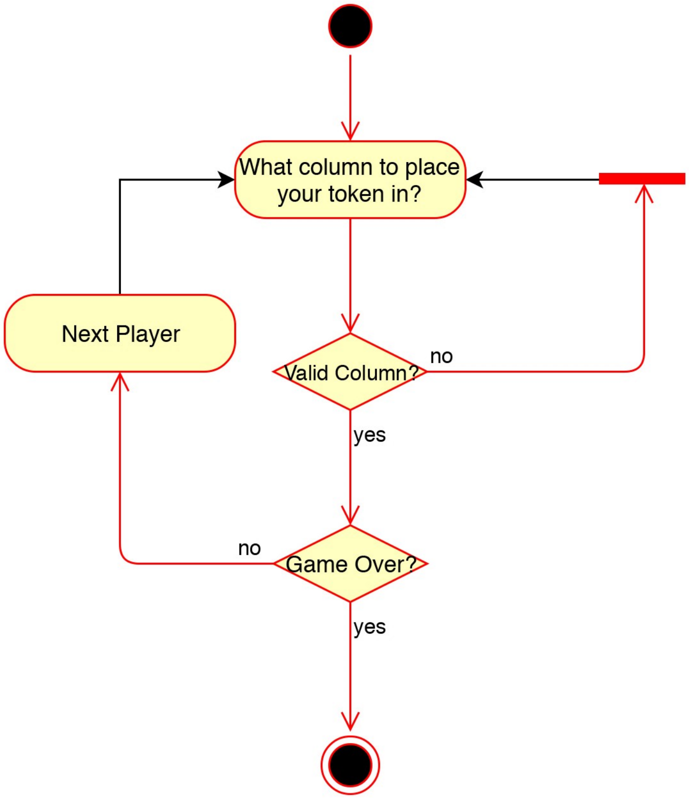


whatsAtPos



Connect4Game

+ main(String [] args):void


What column to place your token in?
Next Player
Valid Column?
no
yes
no
Game Over?
yes