Jeffrey Black
CPSC 2150

How to Play

1. Open a linux terminal and navigate to the folder that contains the folder “cpsc2150” and the file “makefile.”
2. Type “make”
    a. This will compile the java source files, creating .class versions of all of the source files.
3. Type “make run”
    a. This will run the tic-tac-toe program.
    b. Can be ran multiple times.
4. Type “make clean”
    a. This will remove the .class versions of the source files.

Jeffrey Black
CPSC 2150

Requirements Analysis

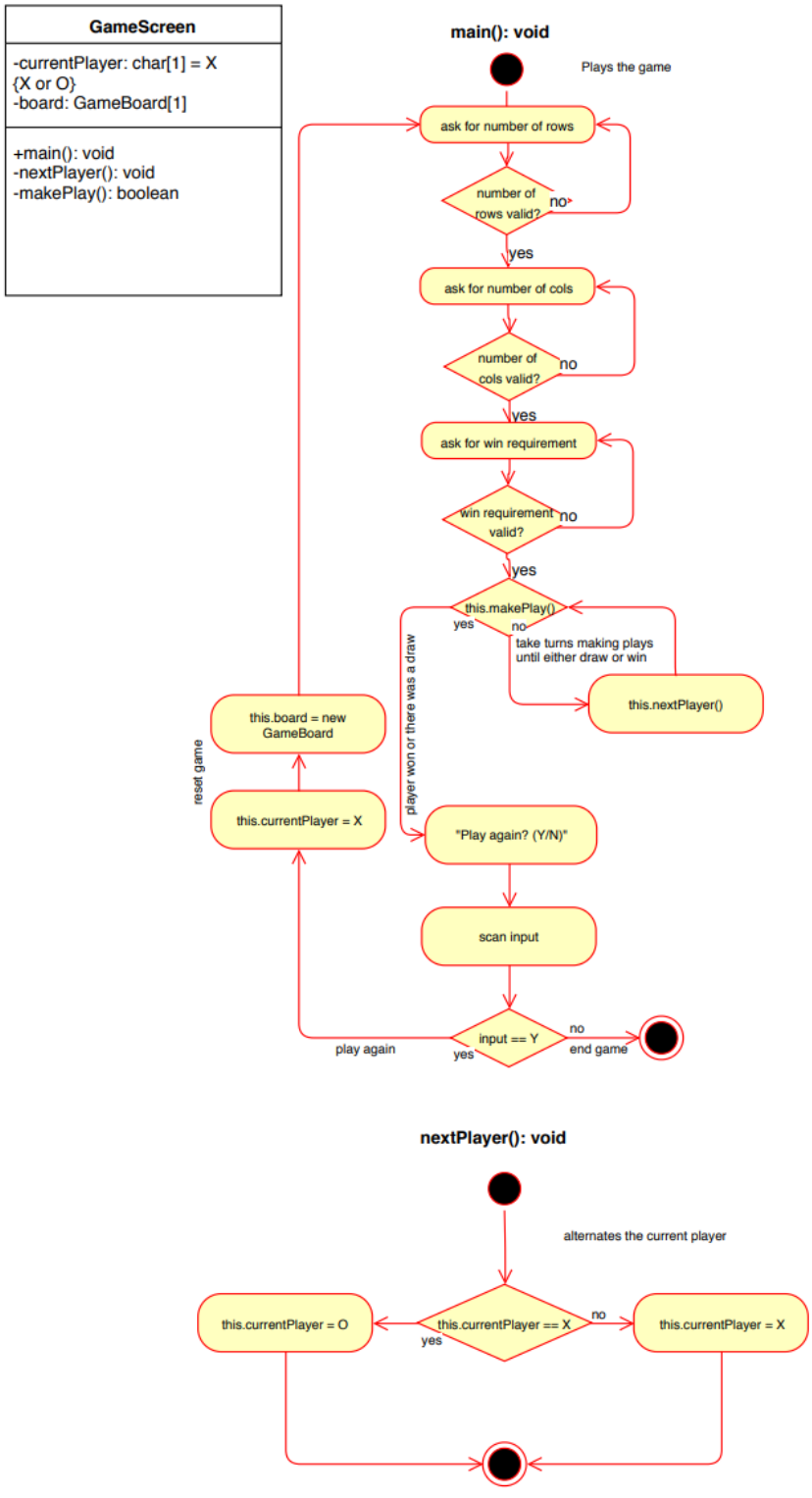
User Stories

- As a player, I want to be able to place an X or O, so I can play Tic-Tac-Toe
- As a player, I want to be able to see the tic-tac-toe grid, so I can view the locations of the X and O's
- As a player, I want to know when one of the players has won, so I don't have to check after each turn
- As a player, I want to know if there has been a tie, so I don't have to check if there has been a tie
- As a player, I want to know whose turn it is, so I know who has to place an X or O
- As a player, I want to know if I placed my marker on a spot that was already claimed, so that both players don't place their marker on the same spot.
- As a player, I want to know if I placed my marker outside of the grid, so I don't place my marker outside of the playable grid
- As a player, I want to be able to play again after the game ends, so I can play more games without starting the program again
- As a player, I want to be able to change the size of the board, so I can play with different board sizes
- As a player, I want to be able to change the number of markers in a row required to win, so I can change the win condition
- As a player, I want to be able to change the game rulesets after choosing to play again, so I can change the rules without restarting the program.

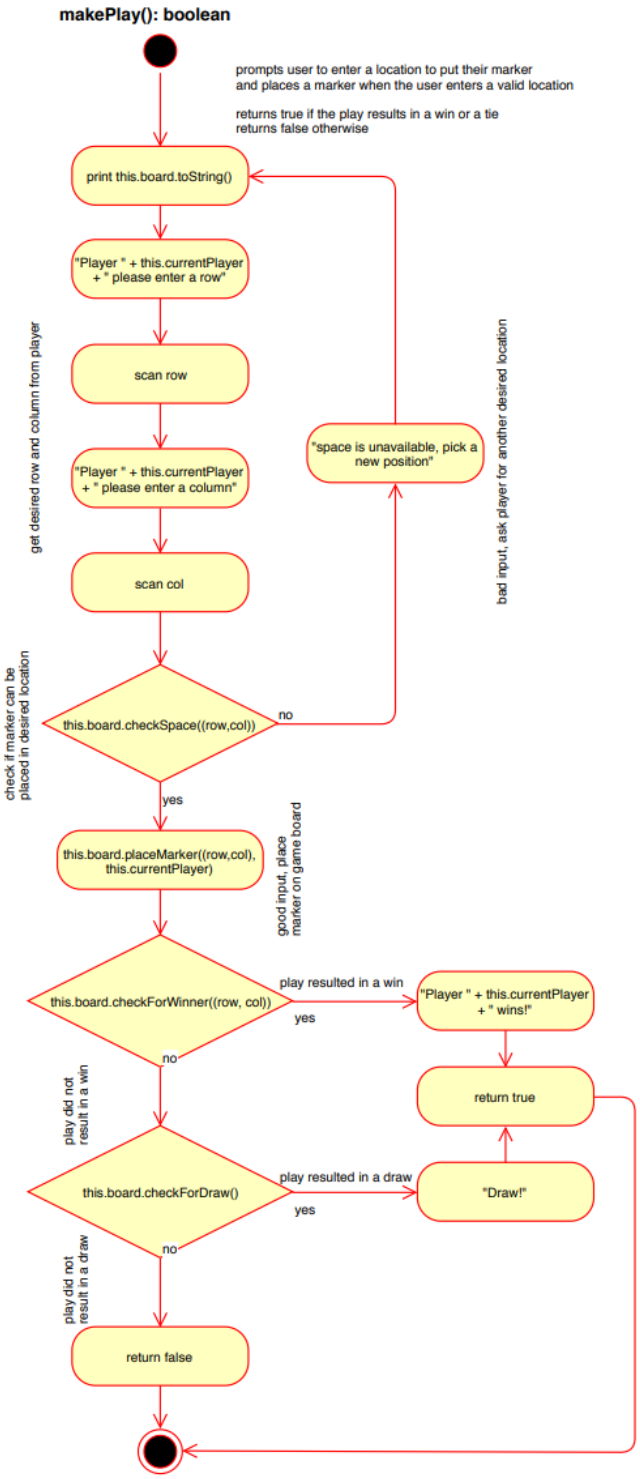
Non-Functional Requirements

- Must have a grid
- X always goes first
- The top left of the board is 0,0
- System must be coded in Java
- System must be able to run on Unix

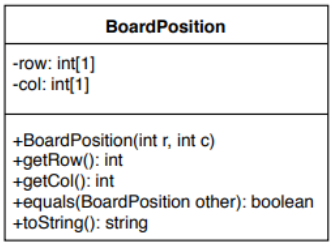
Jeffrey Black
CPSC 2150



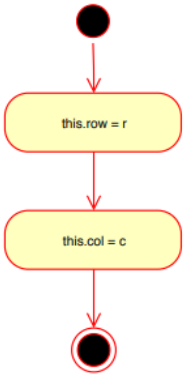
Jeffrey Black
CPSC 2150



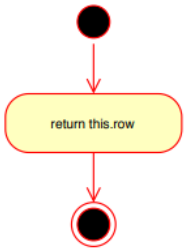
Jeffrey Black
CPSC 2150



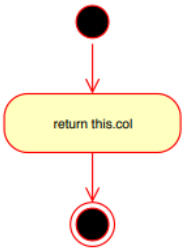
BoardPosition(int r, int c)



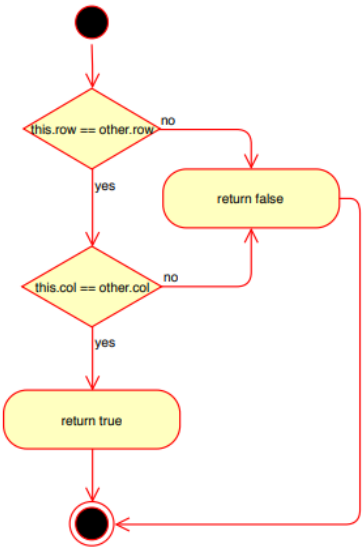
getRow(): int



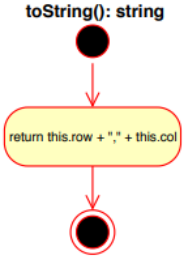
getCol(): int



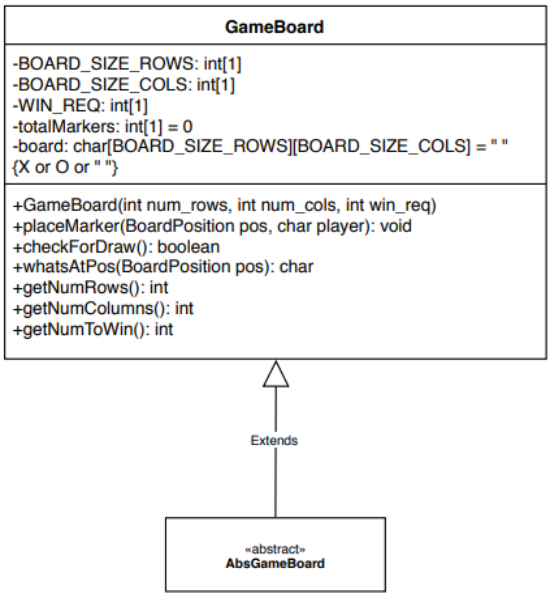
equals(BoardPosition other): boolean

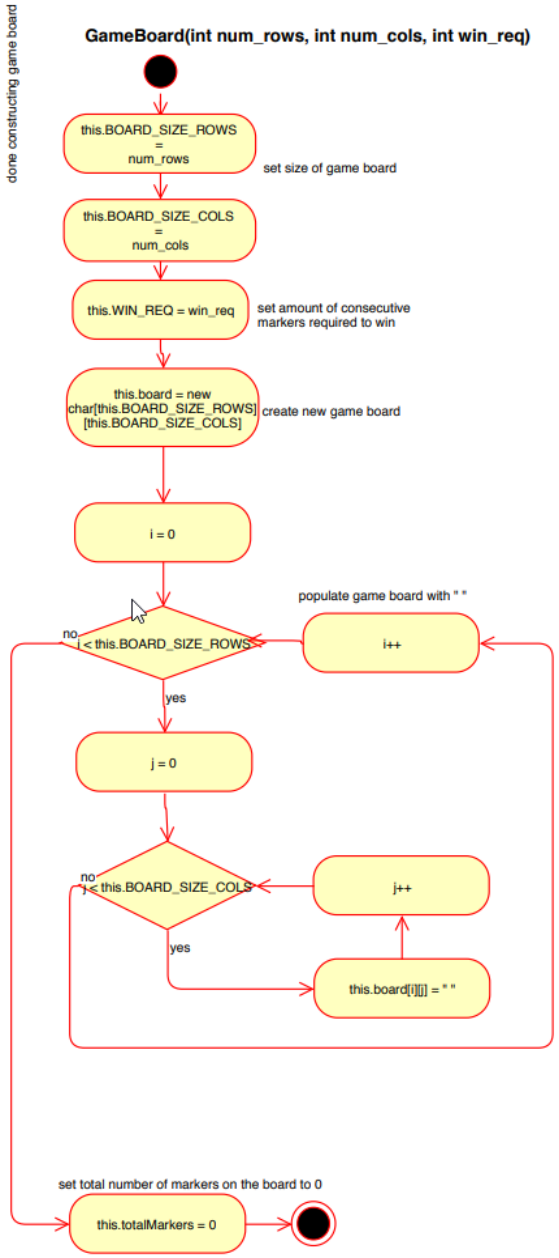


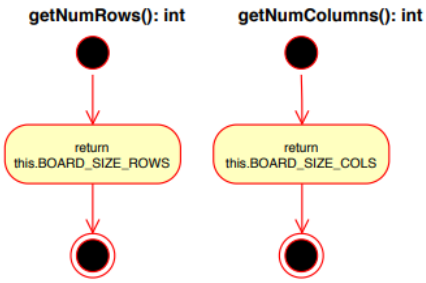
toString(): string

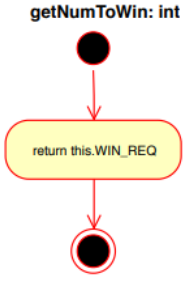


Jeffrey Black
CPSC 2150



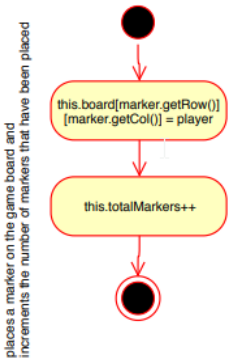




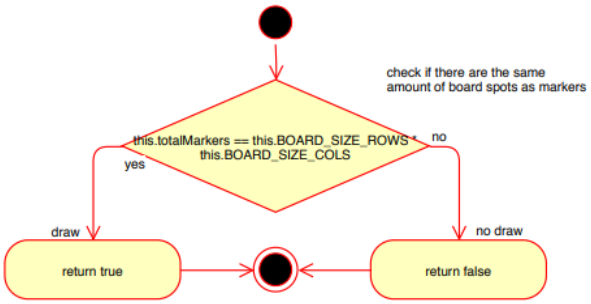


Jeffrey Black
CPSC 2150

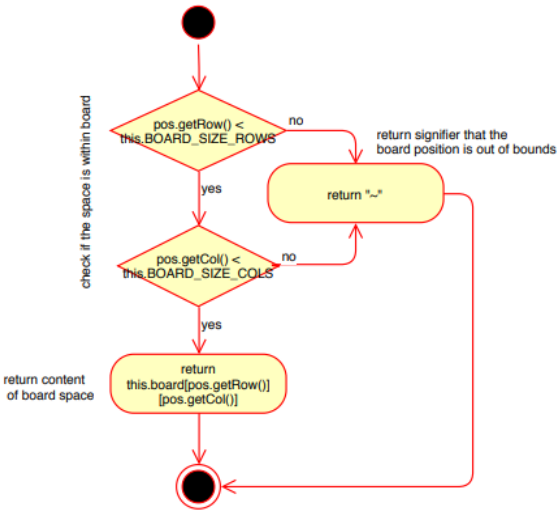
placeMarker(BoardPosition marker, char player): void



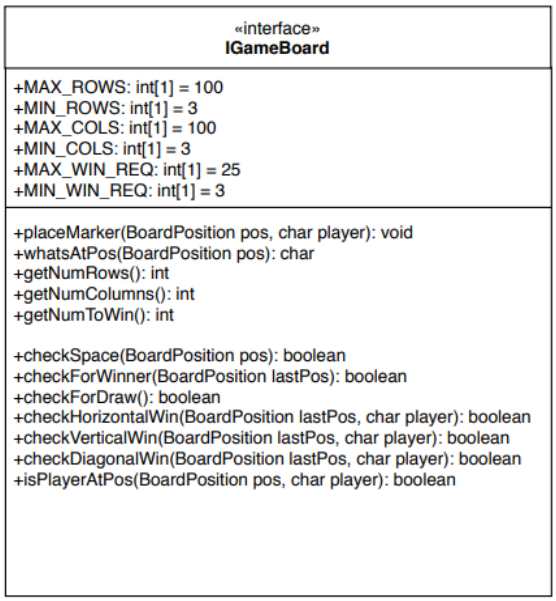
checkForDraw(): boolean



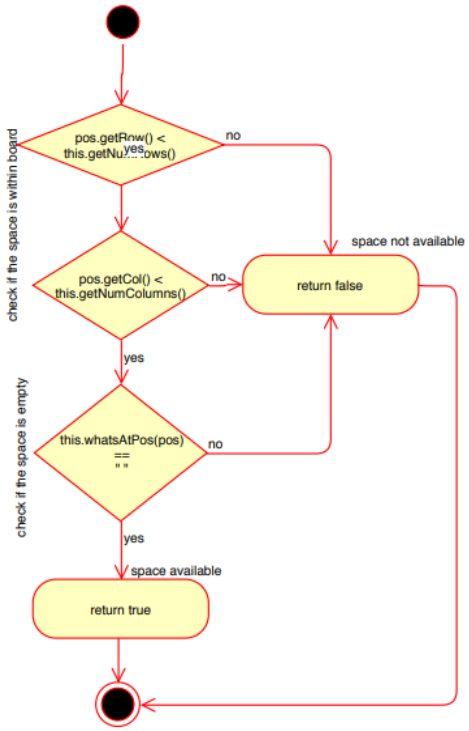
whatsAtPos(BoardPosition pos): char



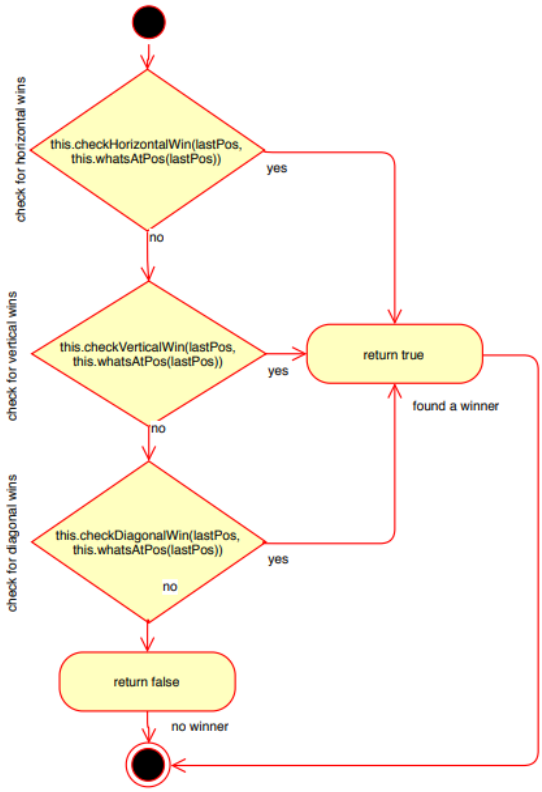
Jeffrey Black
CPSC 2150



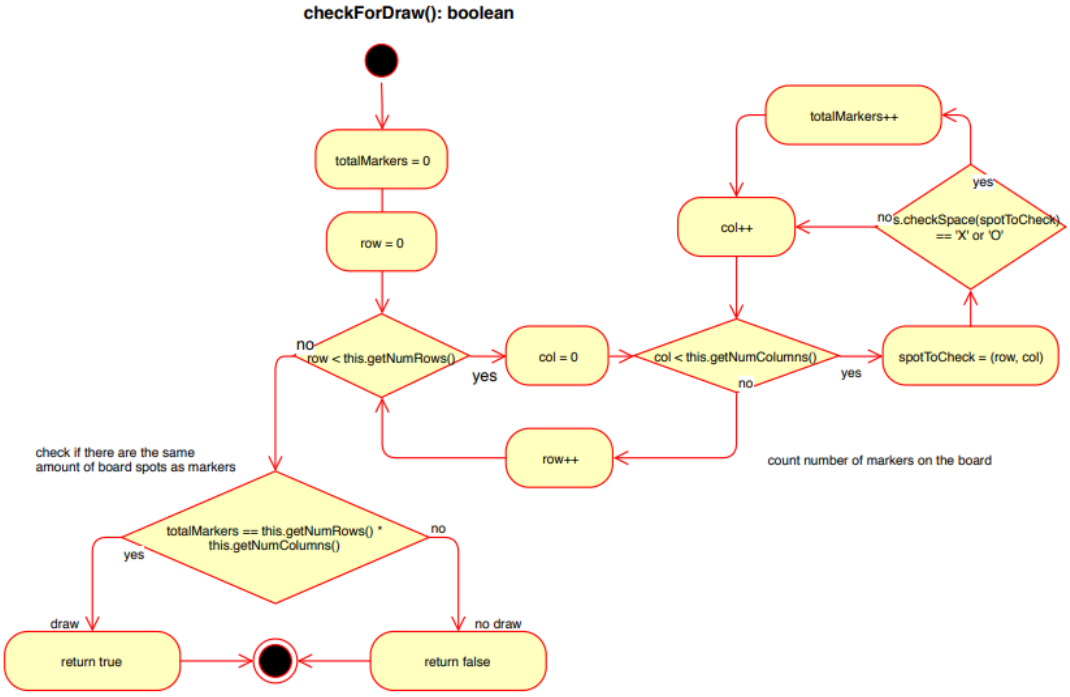
checkSpace(BoardPosition pos): boolean

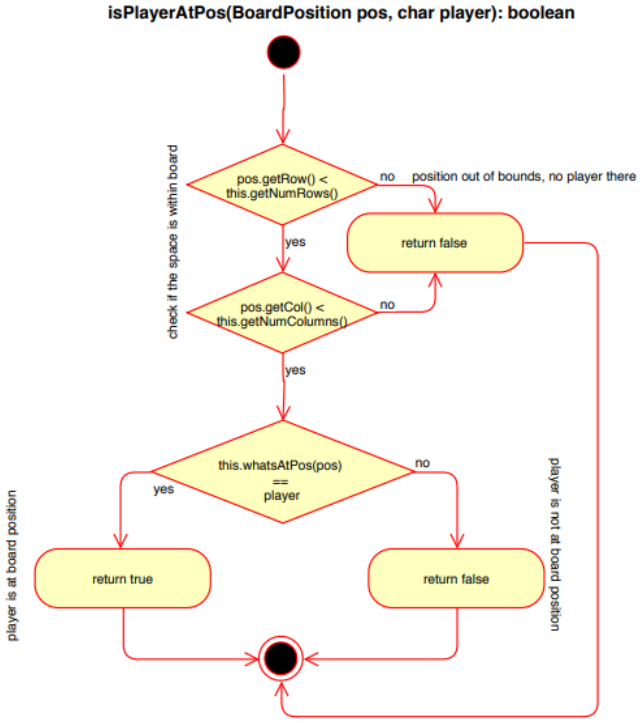


checkForWinner(BoardPosition lastPos): boolean

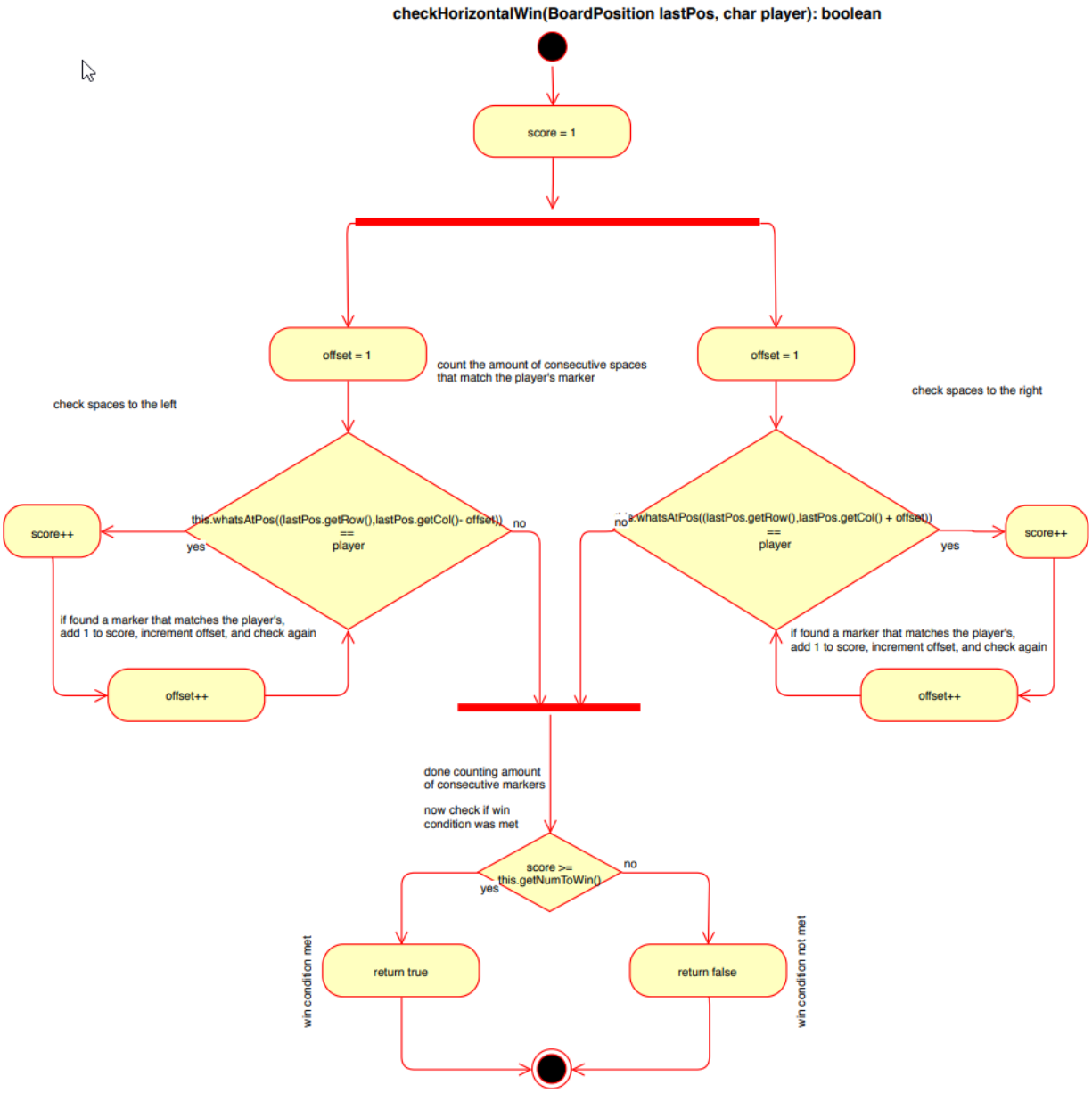


Jeffrey Black
CPSC 2150

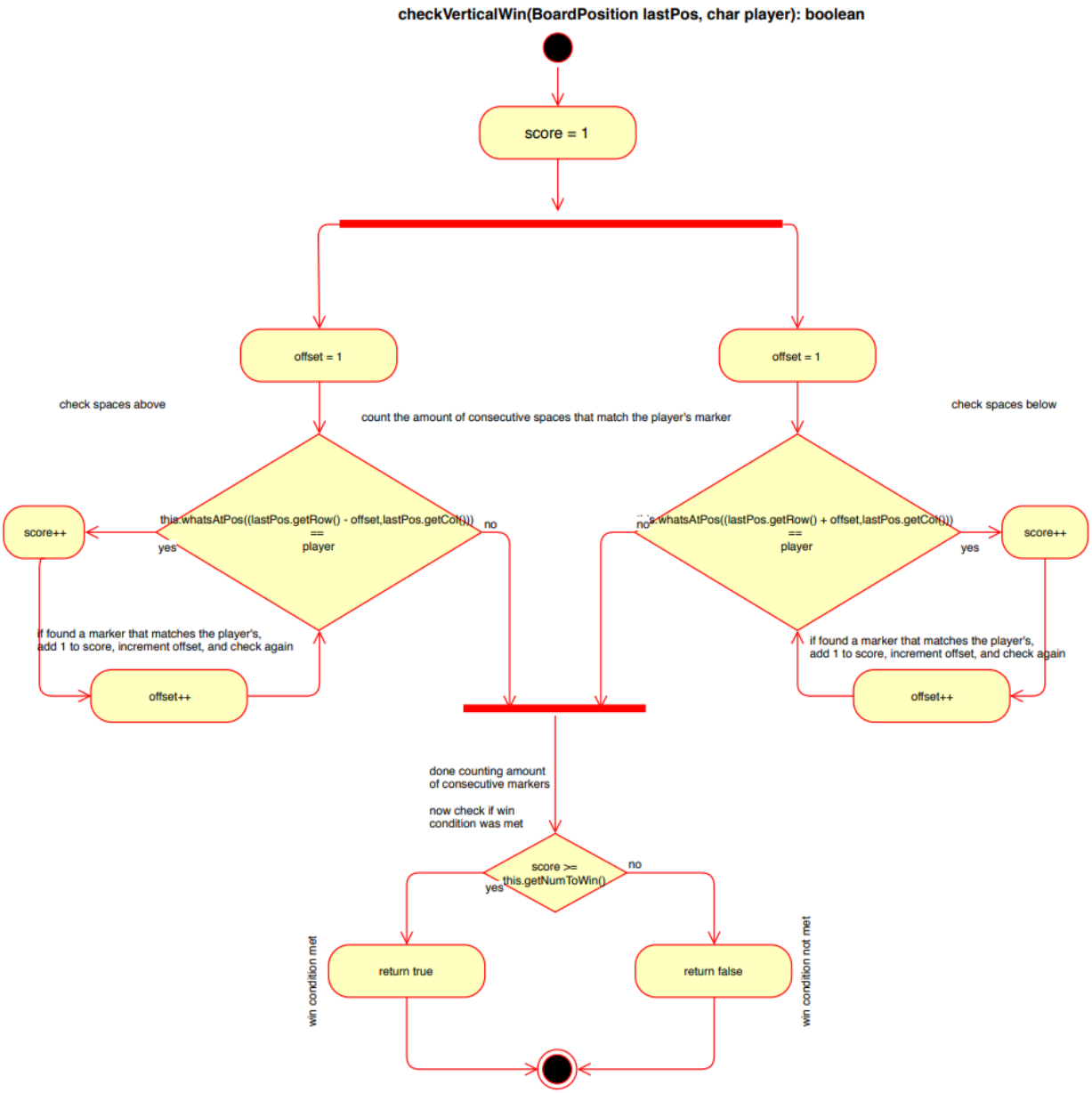




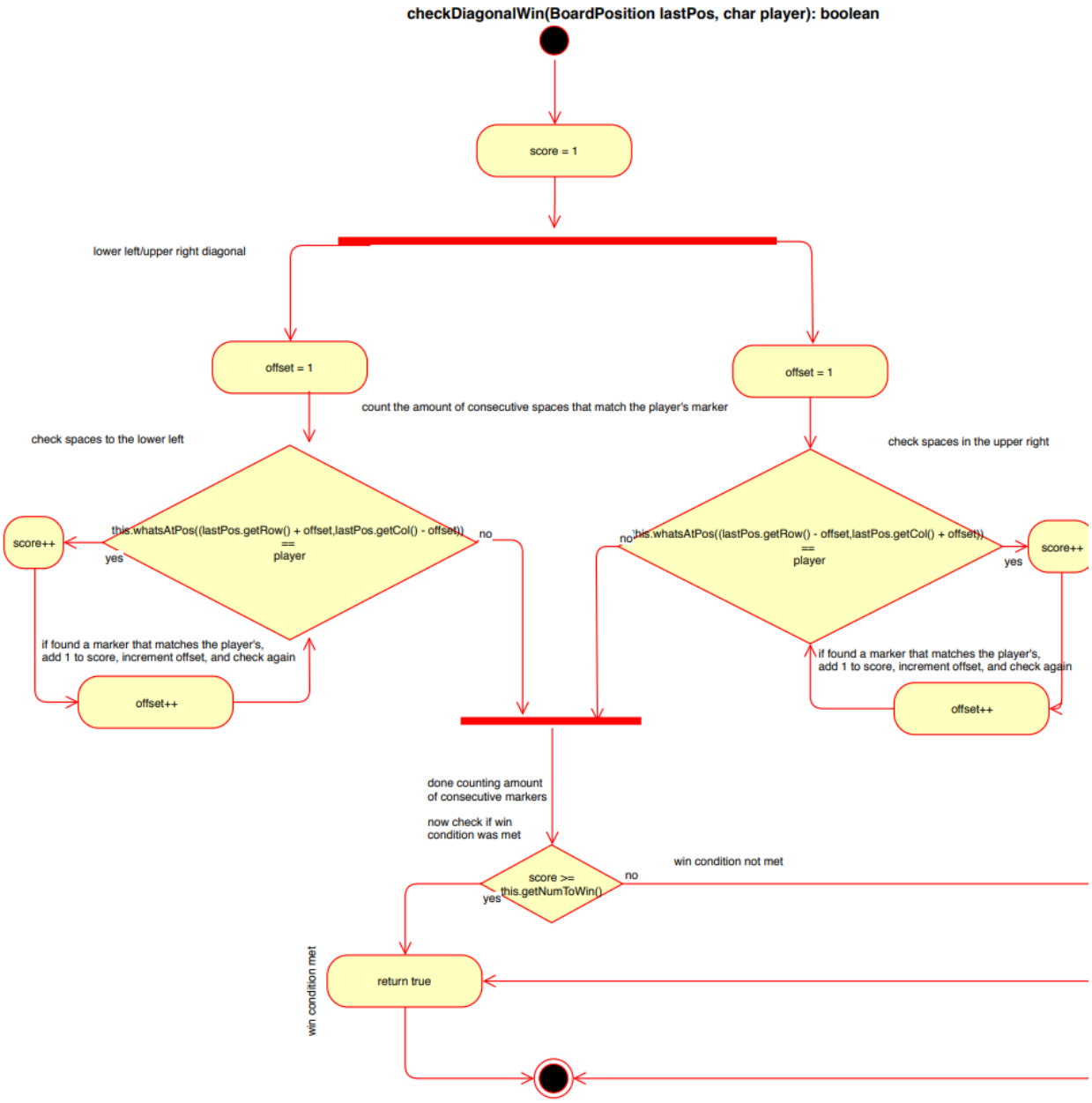
Jeffrey Black
CPSC 2150



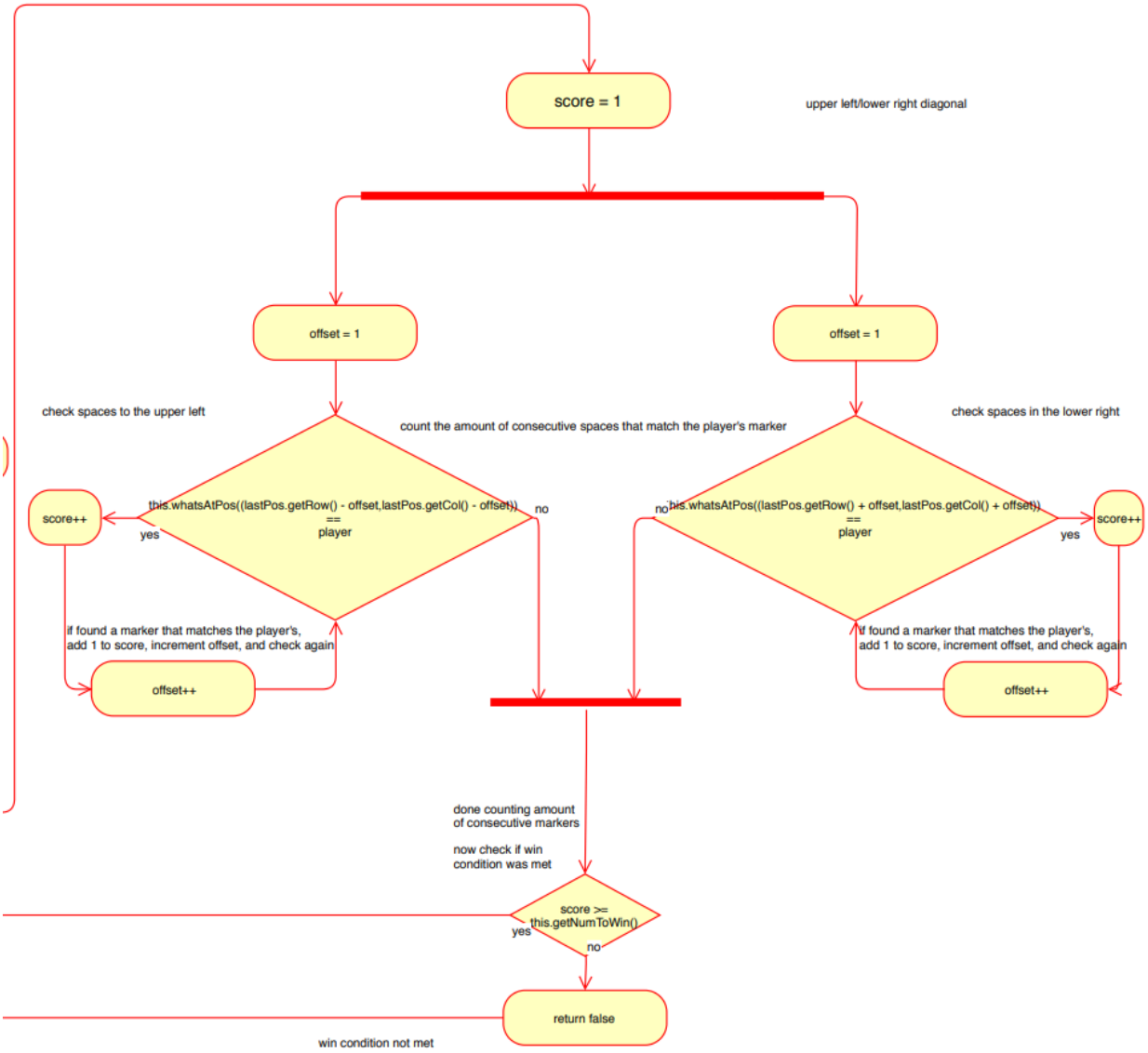
Jeffrey Black
CPSC 2150



Jeffrey Black
CPSC 2150



Jeffrey Black
CPSC 2150



Jeffrey Black
CPSC 2150