Requirement Analysis

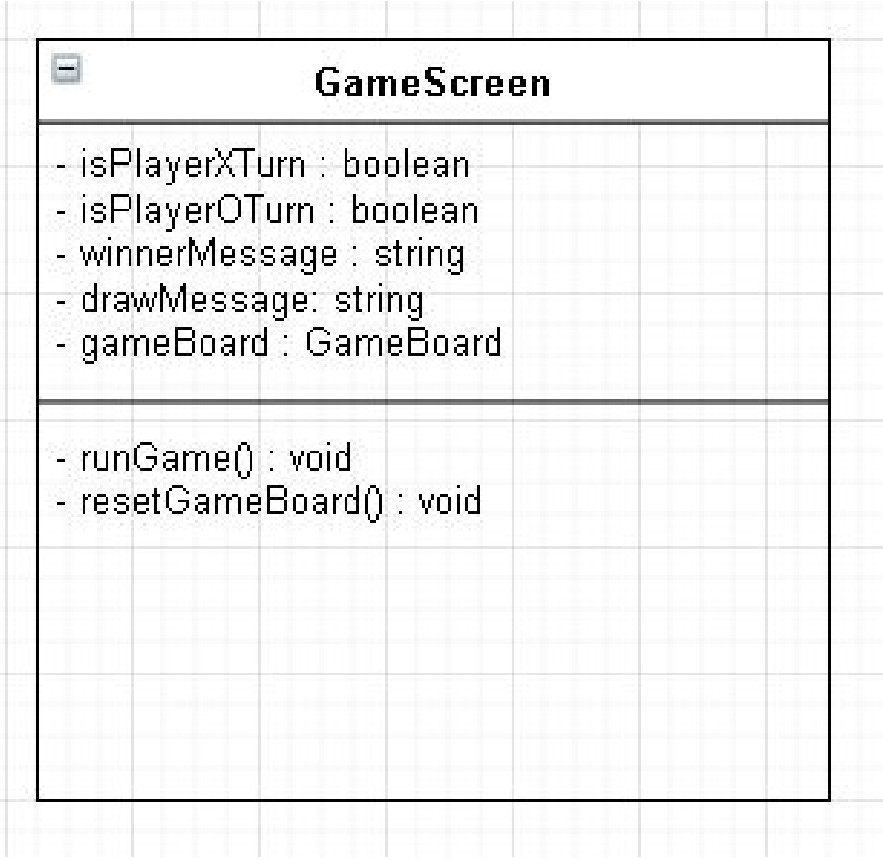
1. As a user I can enter my desired row and column location so my mark can be placed on the board
2. As a user I can play with one more person on the board so I can play with friends
3. As a user I can win the game so I can maybe play again
4. As a user I can tie the game so I can maybe play again
5. As a user I can lose the game so I can restart or play again
6. As a user after I either win lose or tie I can chose to play again or close the game so I can play or go do something else
7. As a user I can see the board printed out so I can determine what my next placement is going to be
8. As a user I can choose to either be X or O by choosing what order I want to go in so that I can be on the side I desire

Nonfunctional Requirements

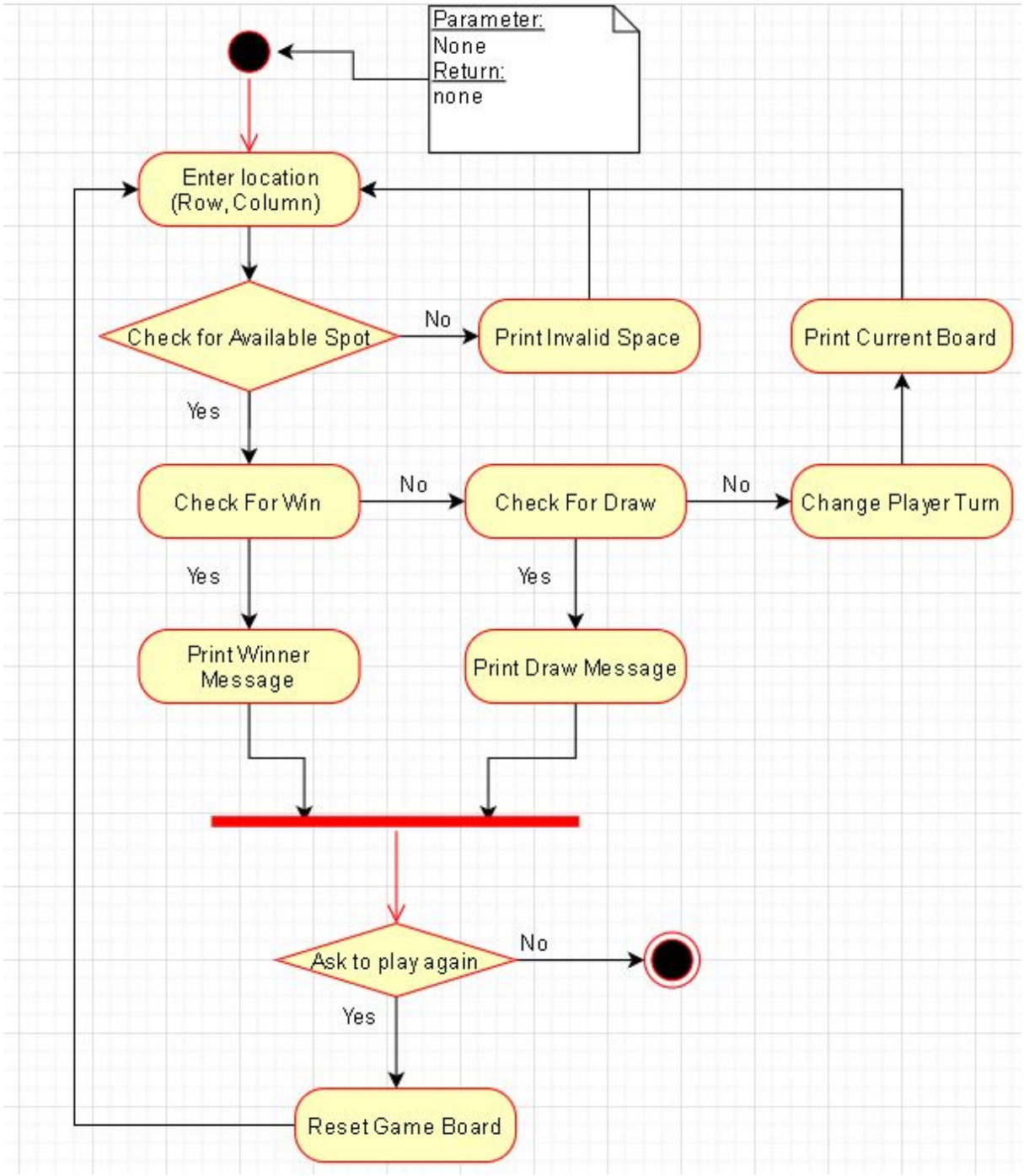
1. Must be written in java
2. Must run on Unix

Design -

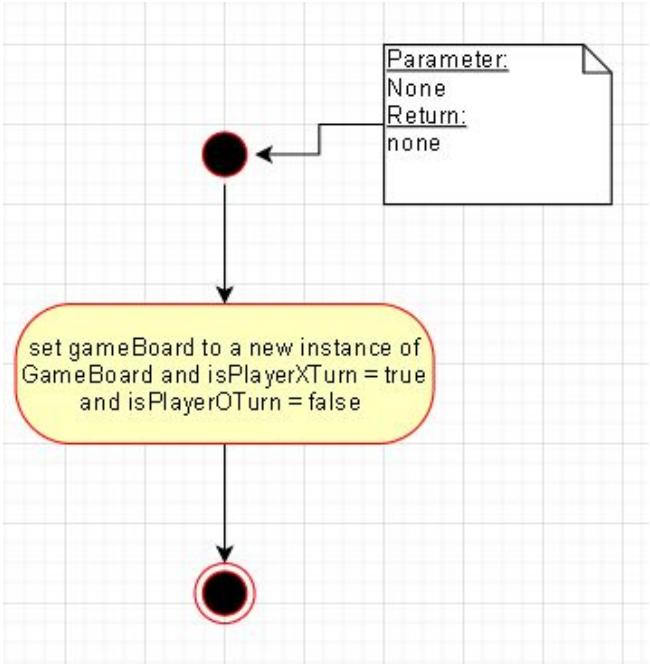
1. Class Diagram of GameScreen



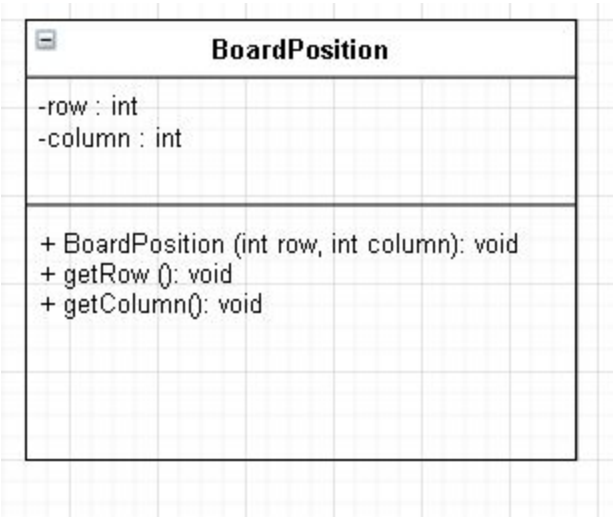
[GameScreen] runGame()


Parameter:
None
Return:
none
Enter location
(Row, Column)
Check for Available Spot
No
Print Invalid Space
Print Current Board
Yes
Check For Win
No
Check For Draw
No
Change Player Turn
Yes
Yes
Print Winner
Message
Print Draw Message
Ask to play again
No
Yes
Reset Game Board

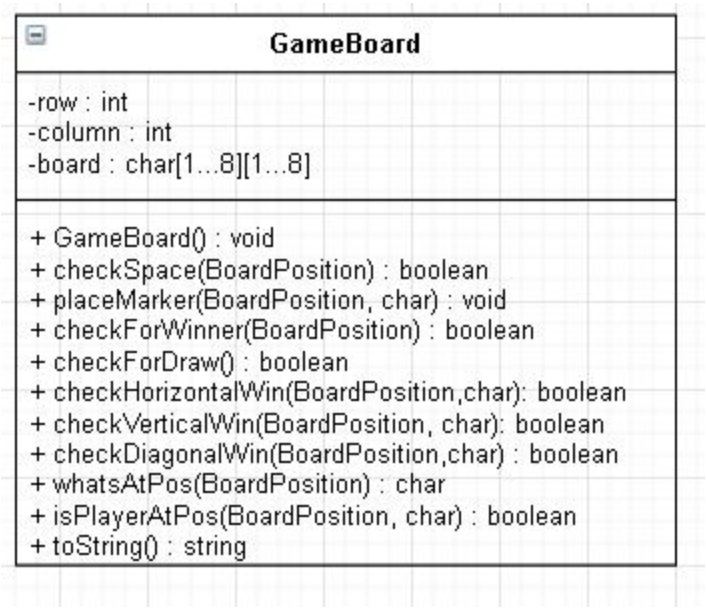
[GameScreen] resetGameBoard()



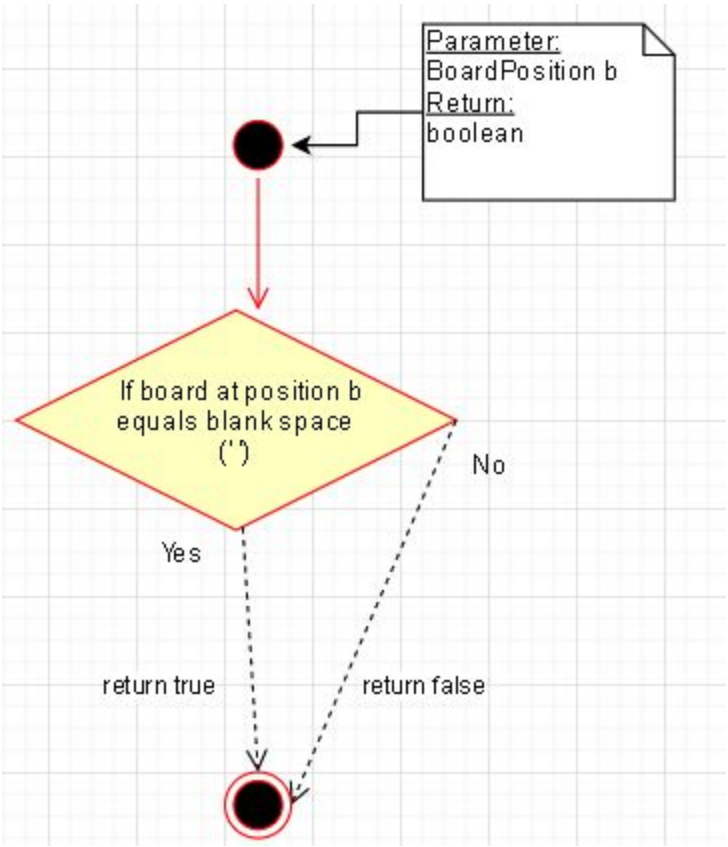
2. Class Diagram of BoardPosition



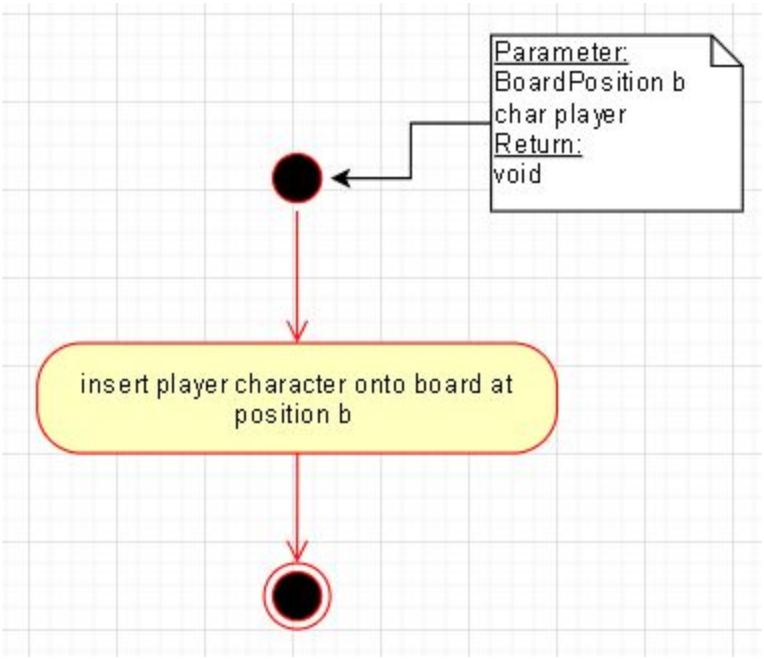
3. Class Diagram of GameBoard



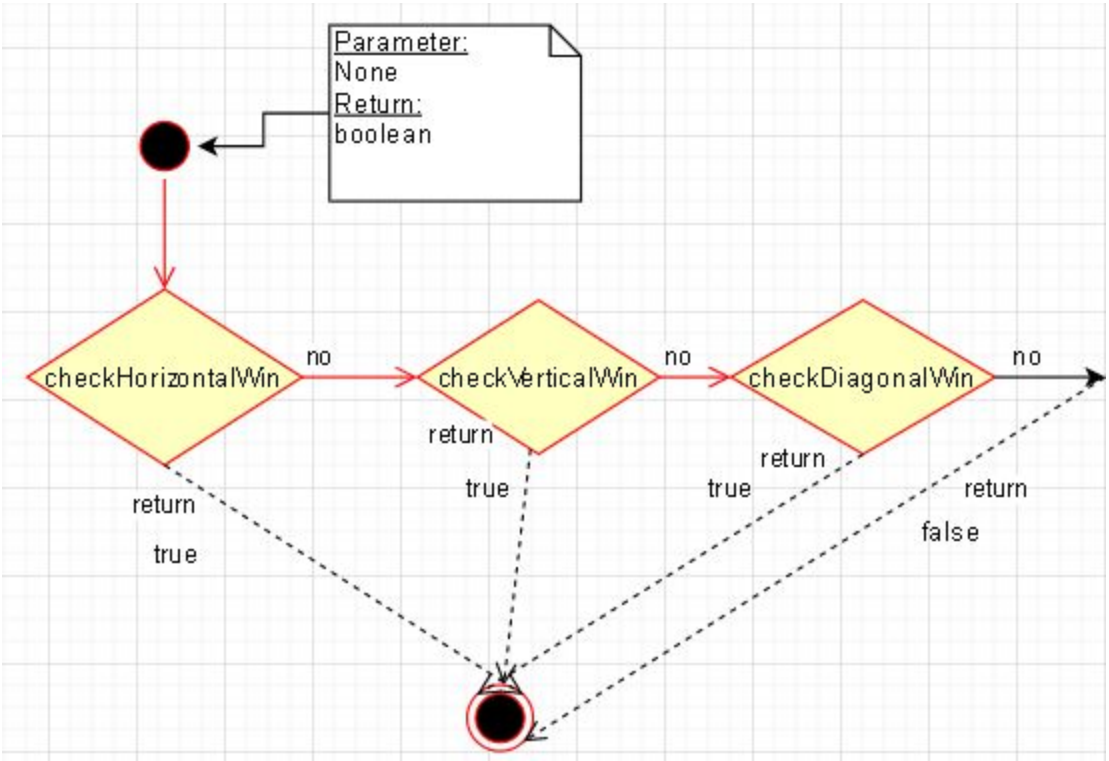
[GameBoard] checkSpace()



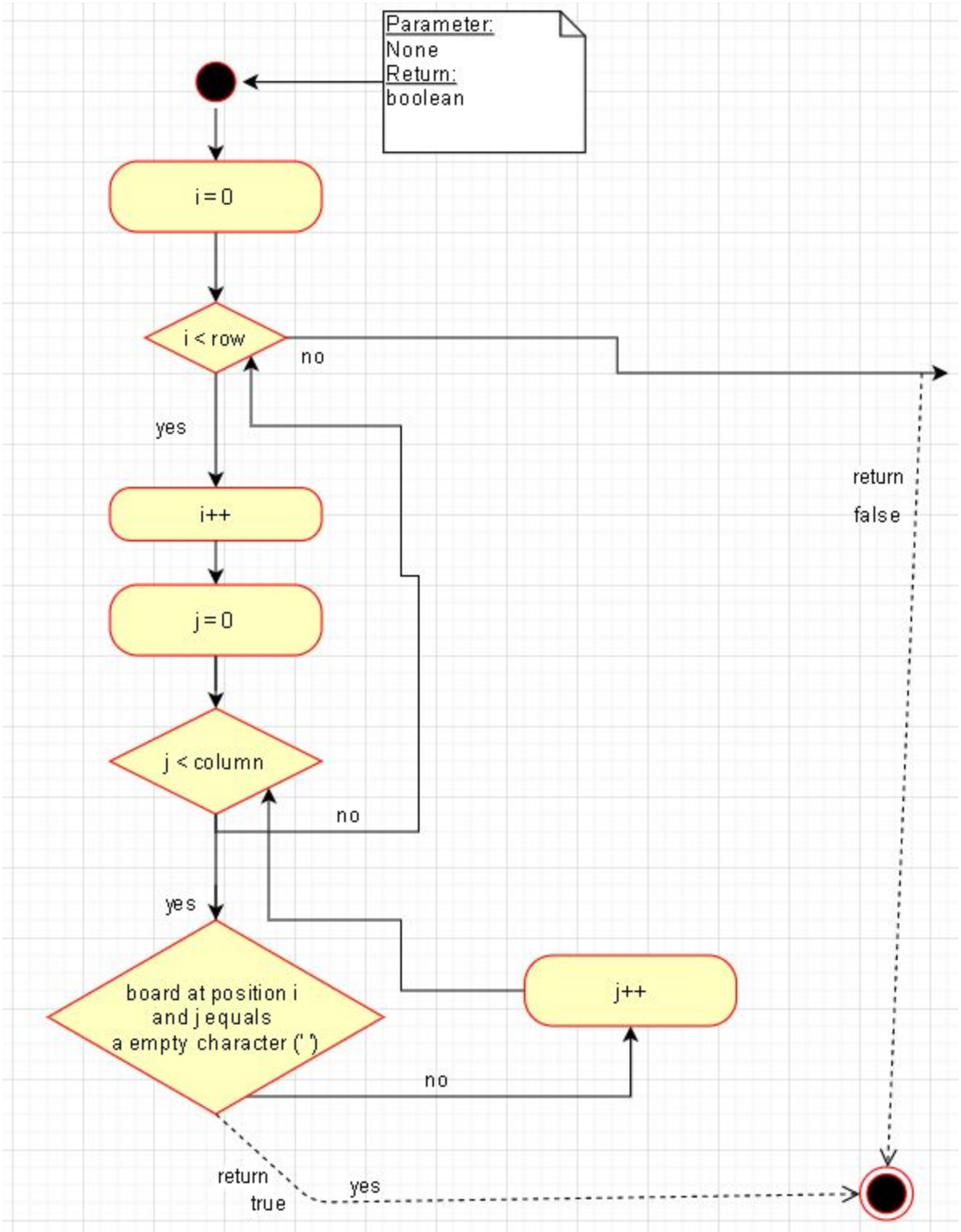
[GameBoard] placeMarker()



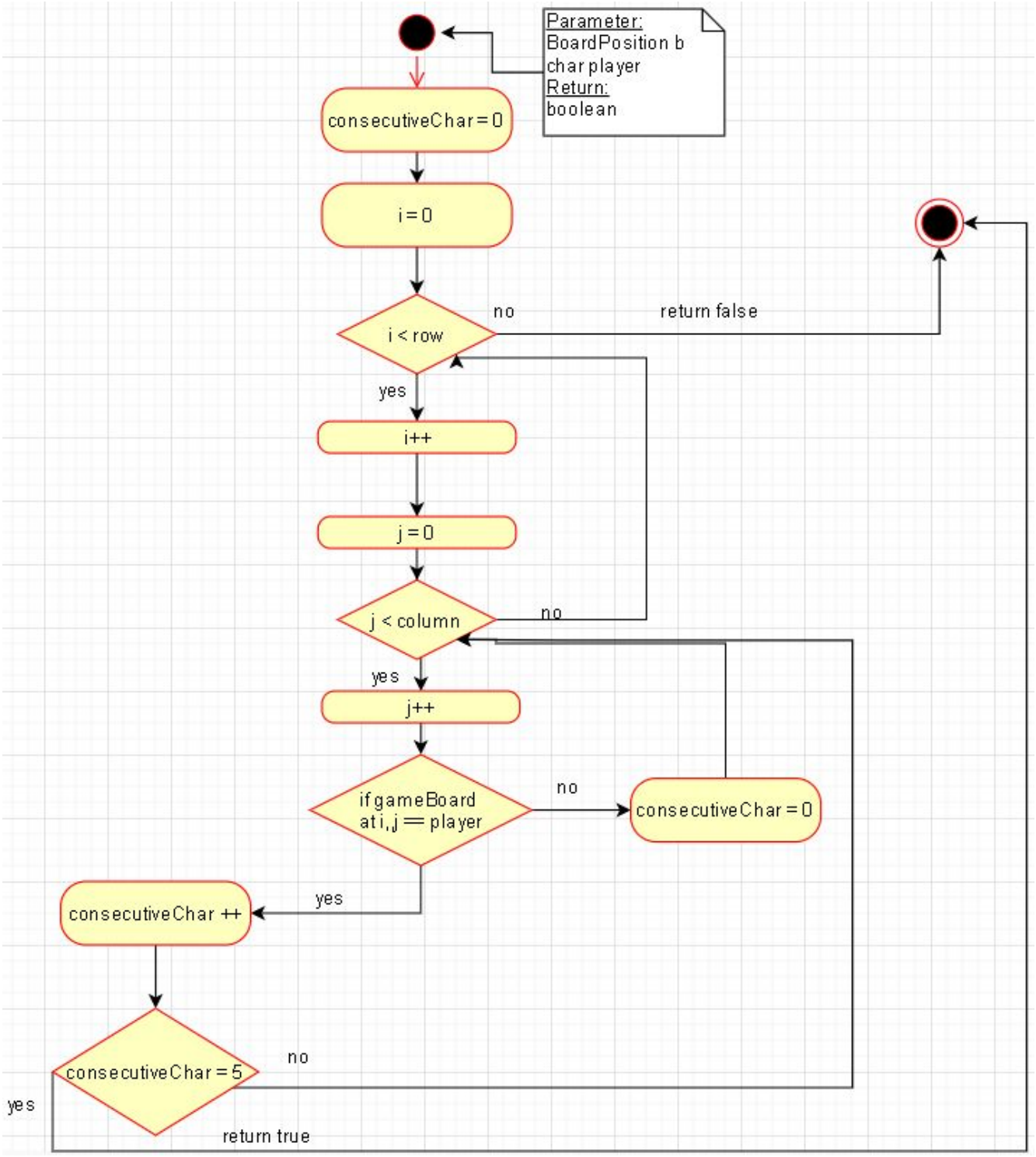
[GameBoard] checkForWinner()



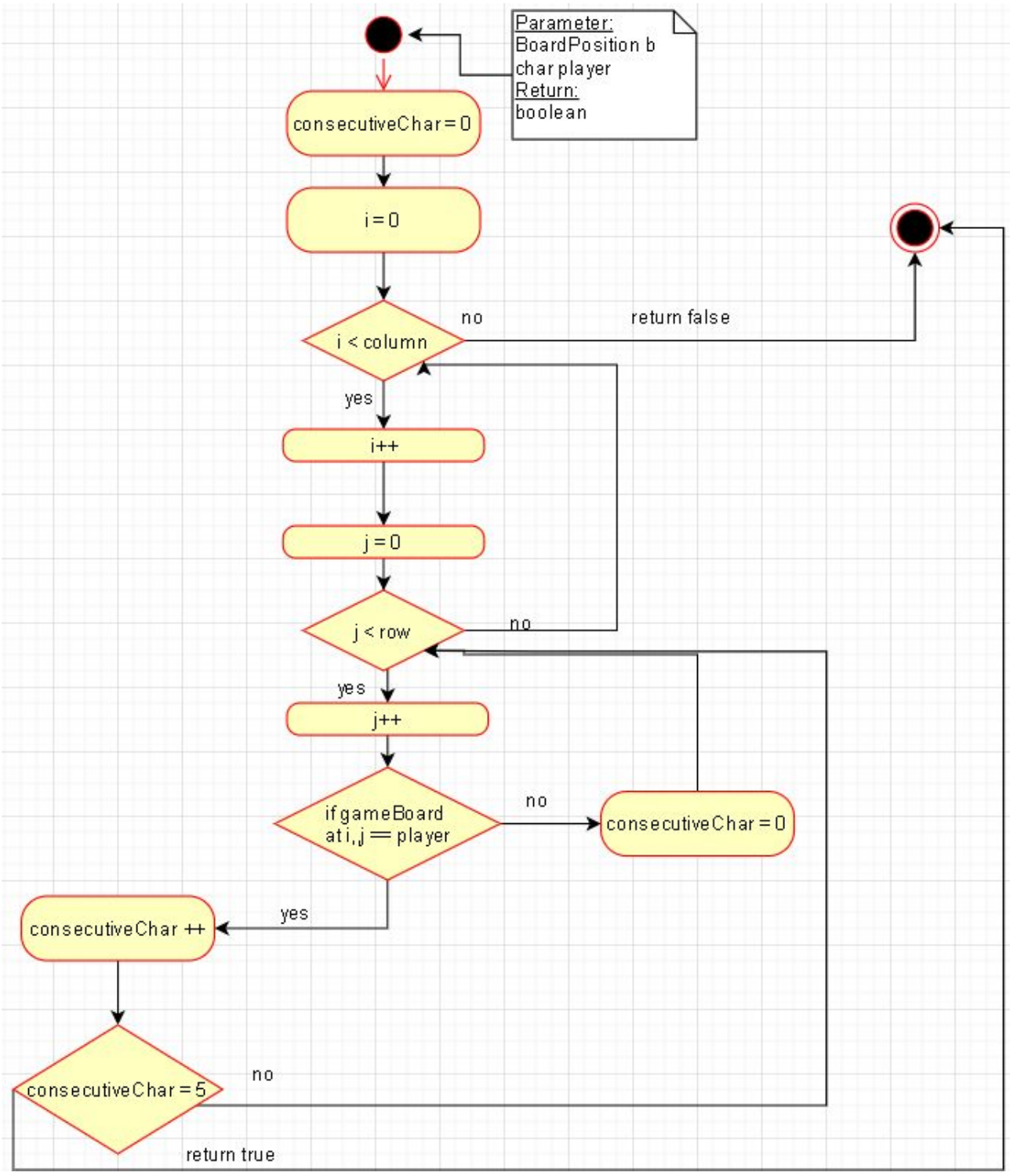
[GameBoard] checkForDraw()



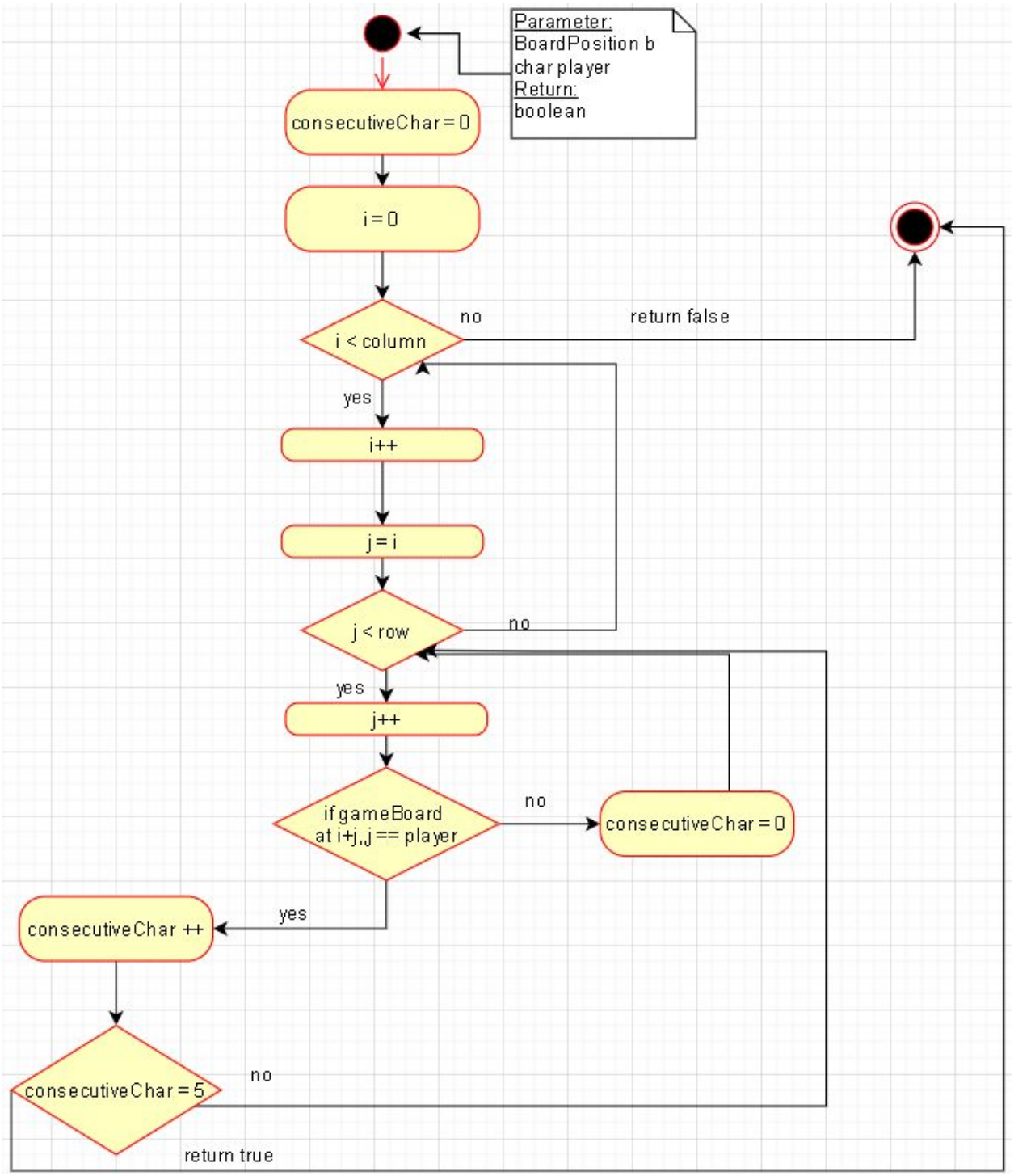
[GameBoard] checkForHorizontalWin()



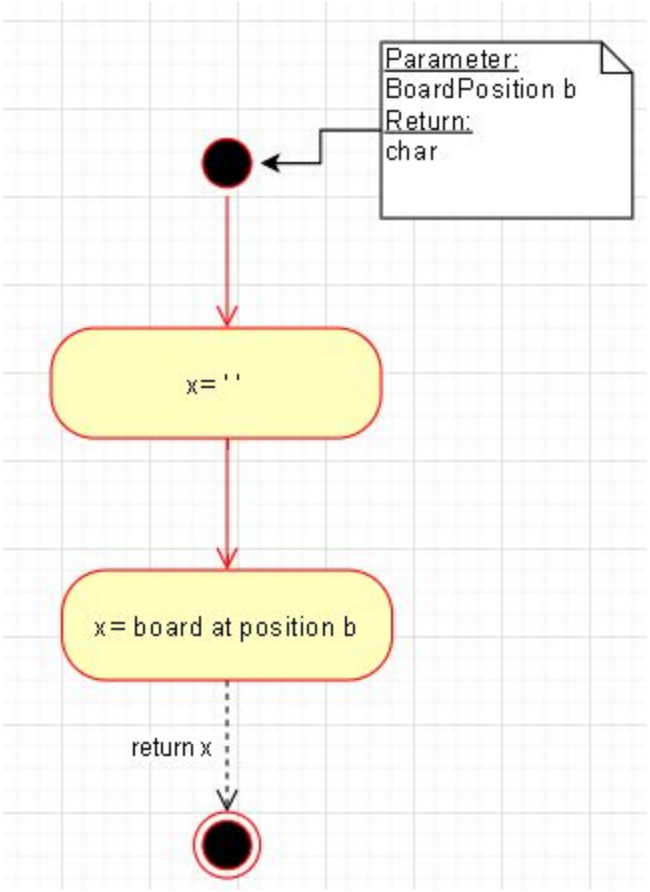
[GameBoard] checkForVerticalWin()



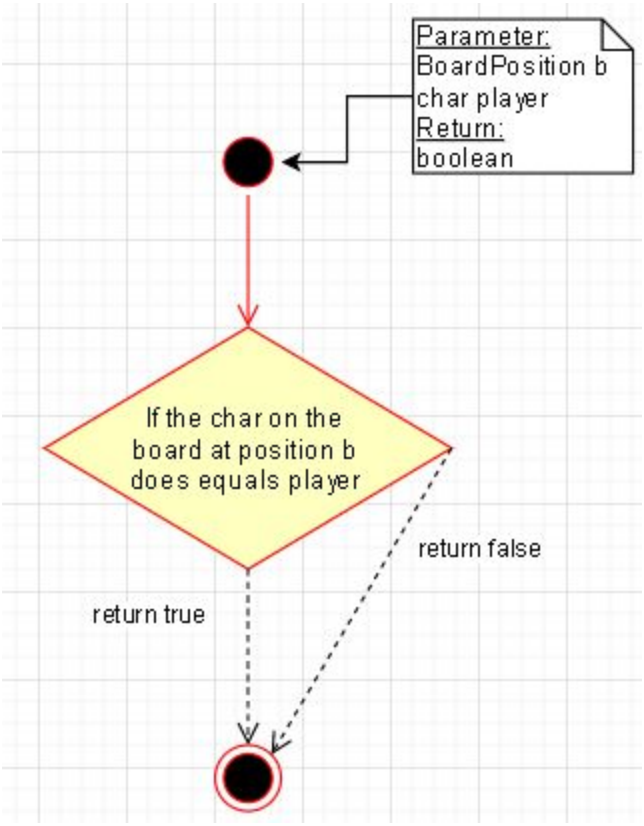
[GameBoard] checkDiagonalWin()



[GameBoard] whatsAtPos()



[GameBoard] isPlayerAtPos()



@override [GameBoard] toString()