

## Requirements Analysis:

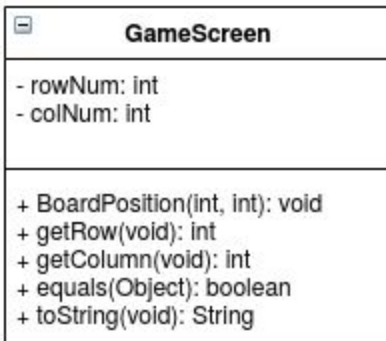
### ➤ Functional Requirements:

- As a player, I need to know whether I am 'X's or 'O's to be able to play the game.
- As a player, I need to be able to place a token in a column of my choice to try and beat my opponent.
- As a player, I need to be able to see the current state of the game board to determine my next move.
- As a player, I need the game to tell me if I select a column that is already full so that I don't break the rules.
- As a player, I need the game to tell me if I select a column that is out of range so that I don't break the rules.
- As a player, I want the game to tell me if I won, lost, or tied with the other player so that I know the result of the game.
- As a player, I want the option to play again if I want to play another game.

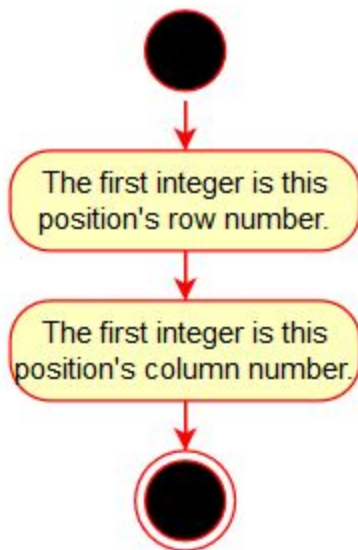
### ➤ Nonfunctional Requirements:

- The game must be programmed using Java.
- The game must run on a Unix system.
- The game must run as efficiently as possible.

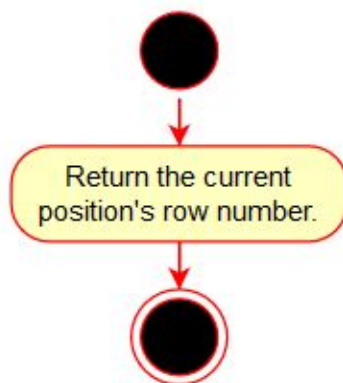
## BoardPosition Class:



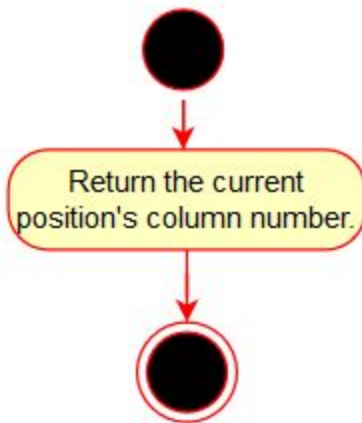
public BoardPosition(int row, int column):



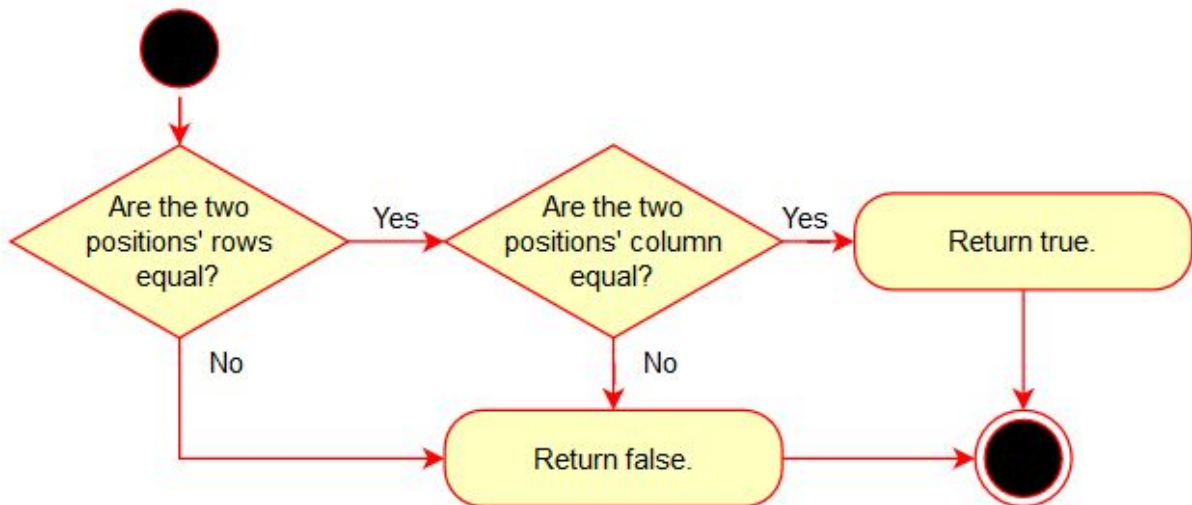
public int getRow():



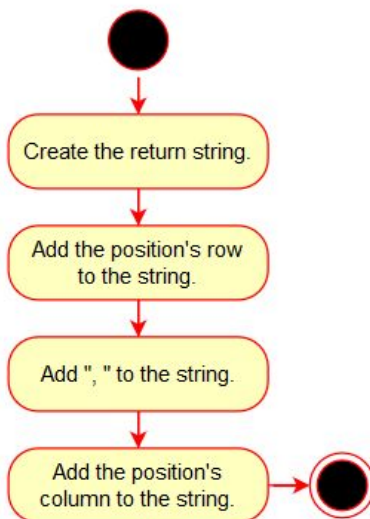
public int getColumn():



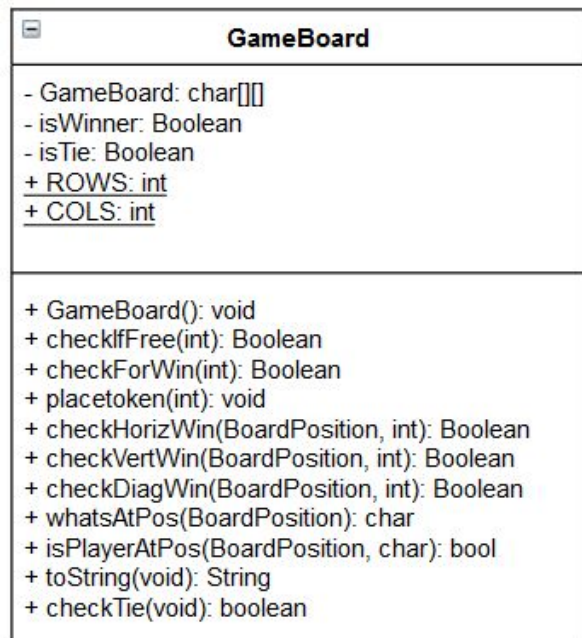
public bool equals():



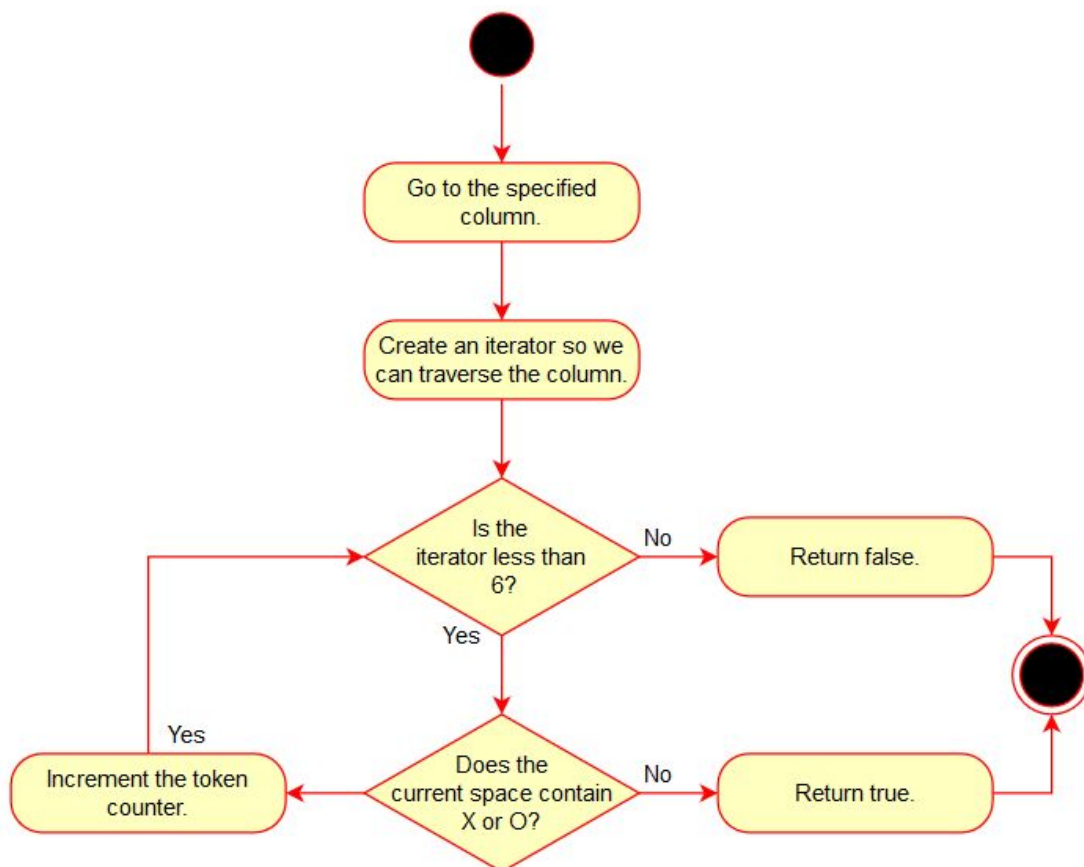
public String toString():



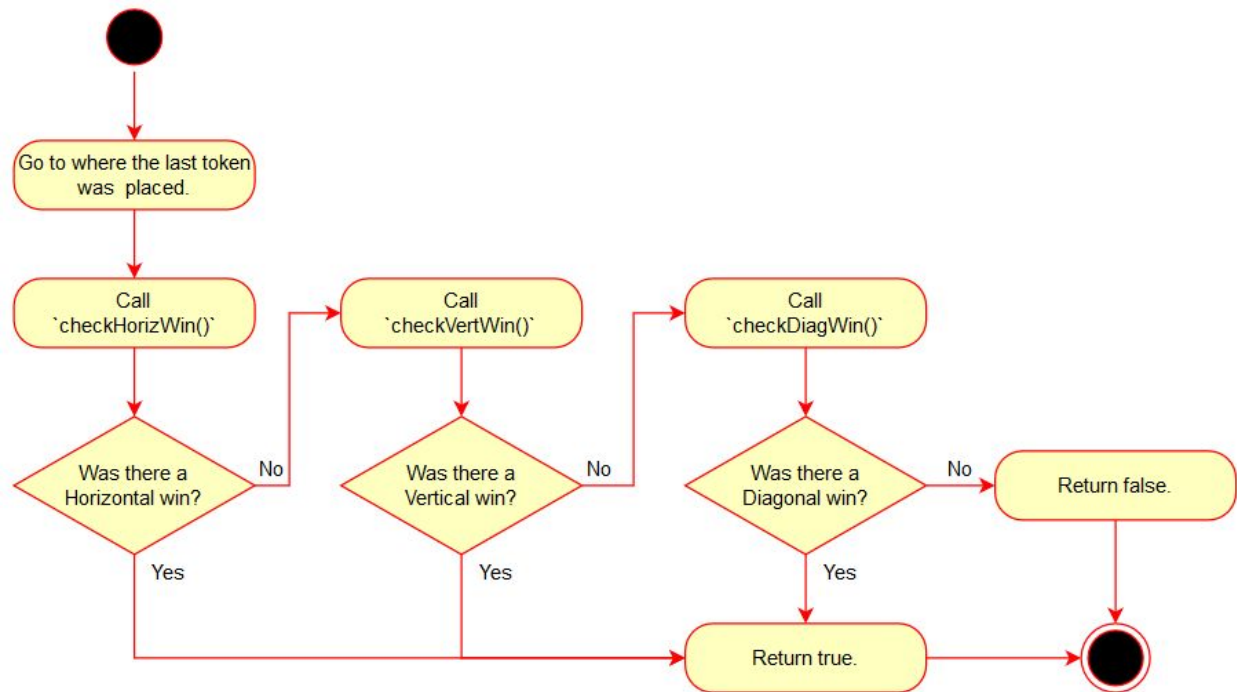
GameBoard class:



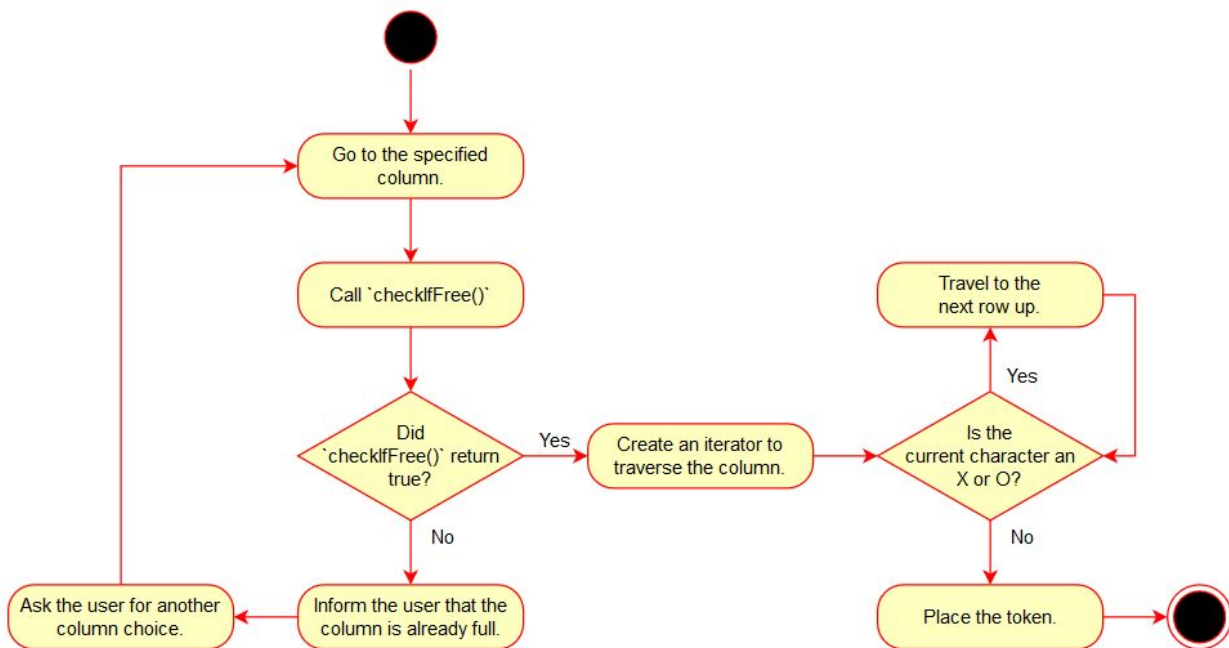
public boolean checkIfFree(int c):



public boolean checkForWin(int c):



public void placeToken(int c):



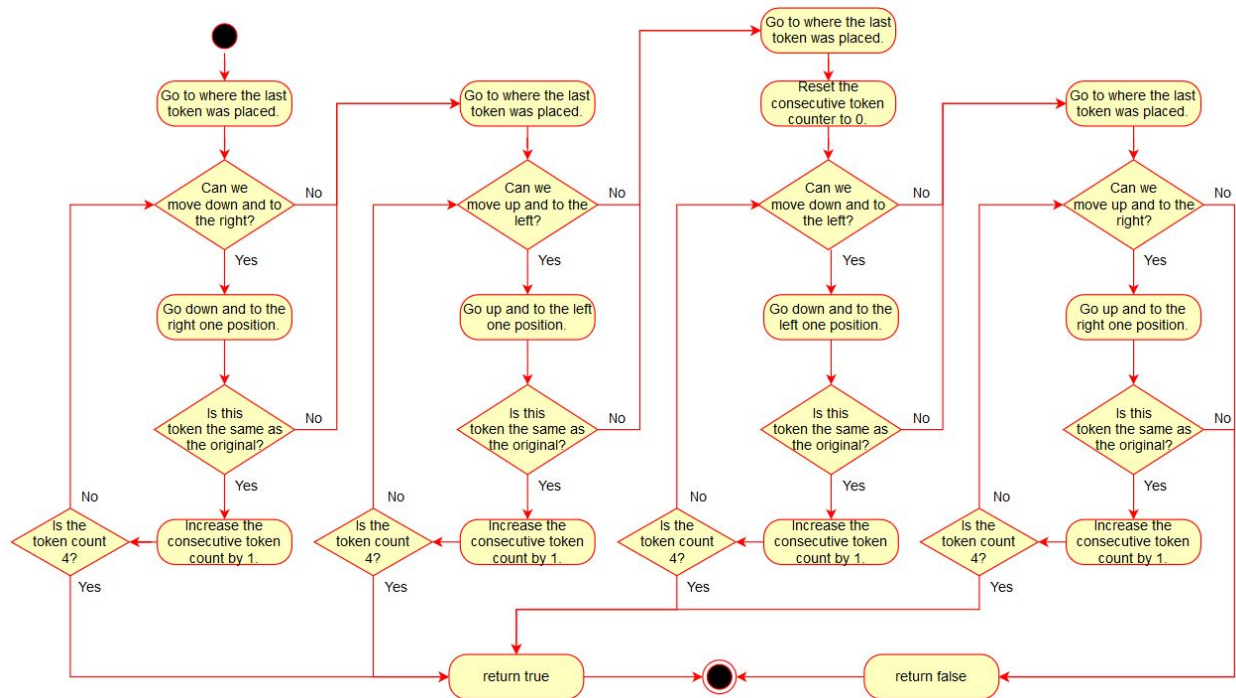
public boolean checkHorizWin(BoardPosition pos, char p):

```

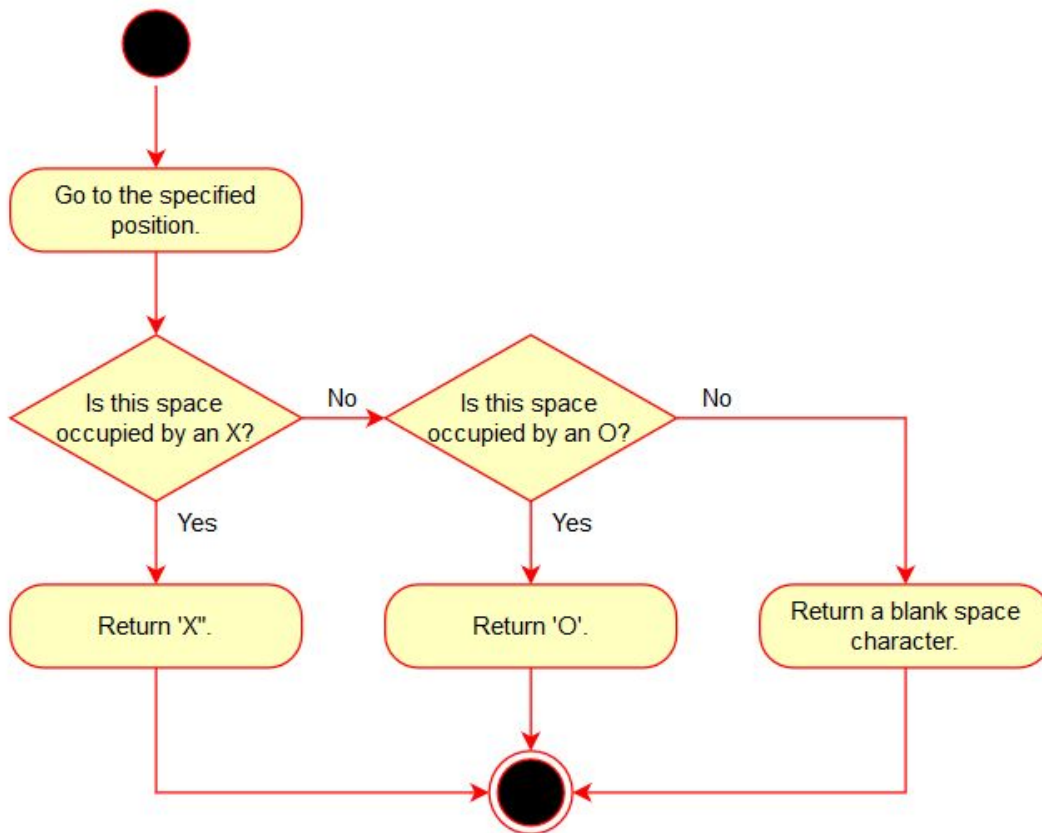
graph TD
    Start(( )) --> MoveDown[Go to where the last token was placed.]
    MoveDown --> StillInBounds{Still in bounds if we move down?}
    StillInBounds -- No --> ReturnFalse[return false]
    StillInBounds -- Yes --> MoveDownOne[Move down one position.]
    MoveDownOne --> IsSameOriginal{Is this token the same as the original?}
    IsSameOriginal -- No --> ReturnFalse
    IsSameOriginal -- Yes --> IncreaseTally[Increase the consecutive token tally.]
    IncreaseTally --> IsConsecutiveCount4{Is the consecutive token count 4?}
    IsConsecutiveCount4 -- No --> StillInBounds
    IsConsecutiveCount4 -- Yes --> ReturnTrue[return true]
    ReturnTrue --> End((( )))
    ReturnFalse --> End

```

public boolean checkDiagWin(BoardPosition pos, char p):

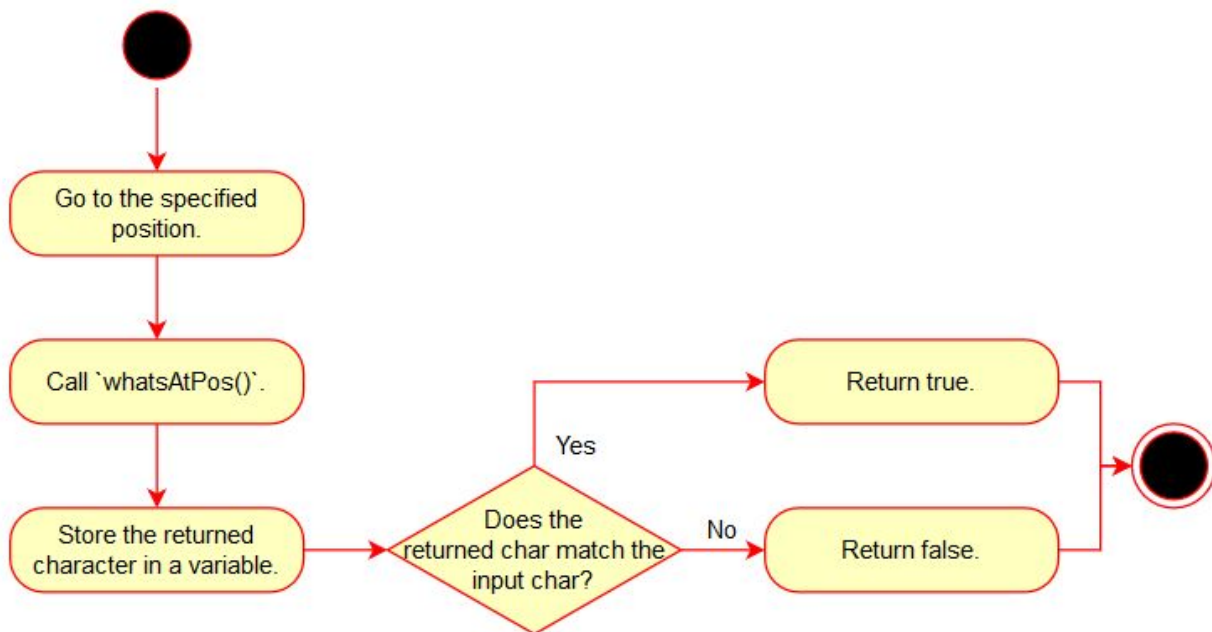


public char whatsAtPos(BoardPosition pos):

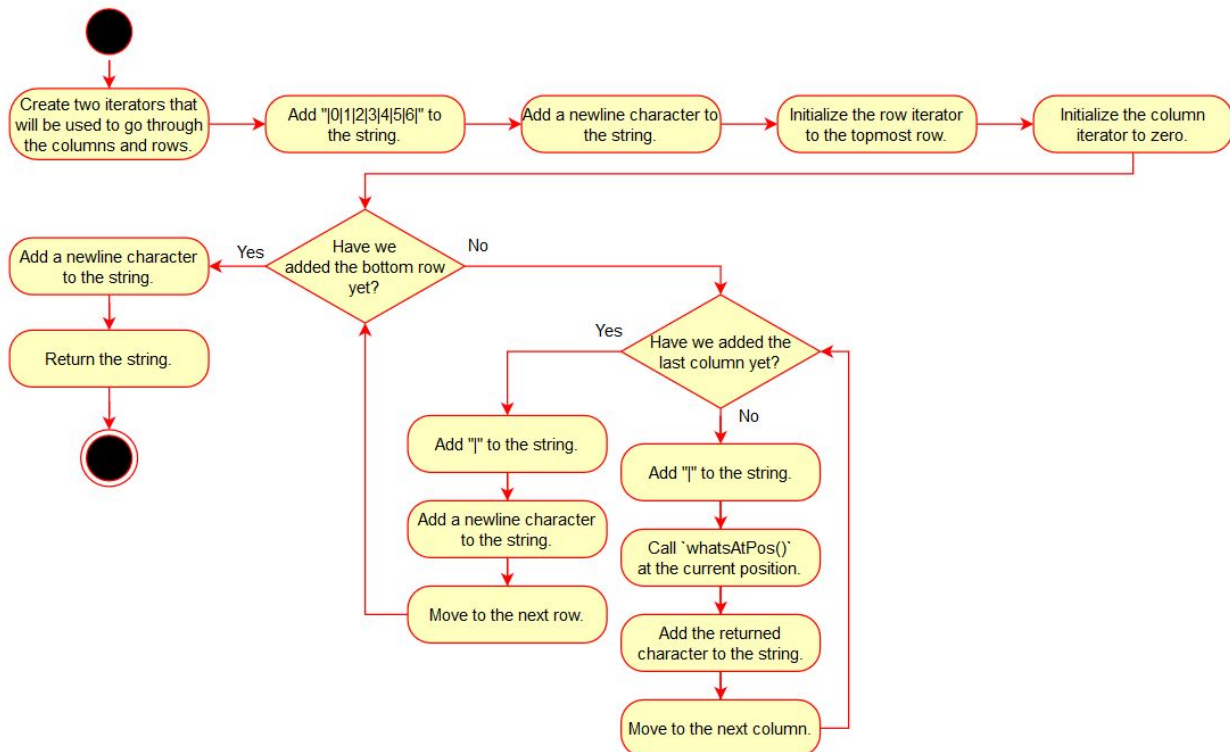




public boolean isPlayerAtPos(BoardPosition pos):

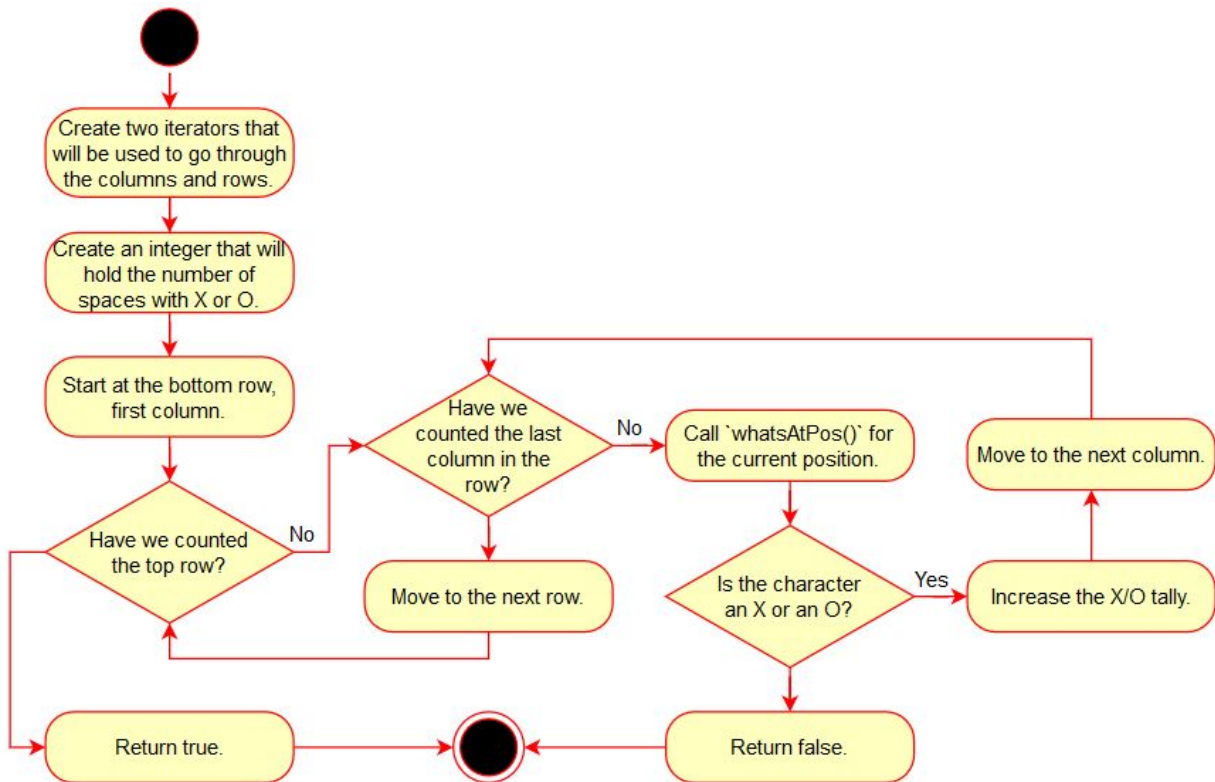


String toString():

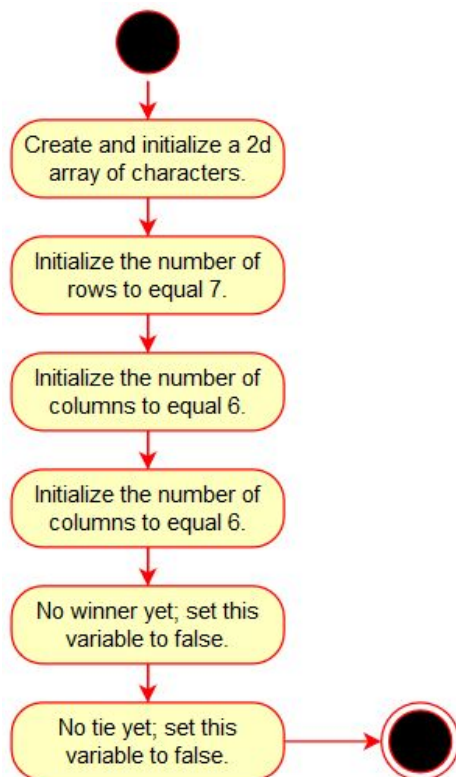




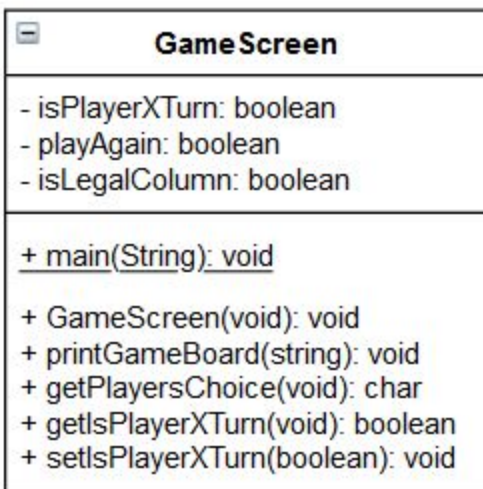
boolean checkTie():



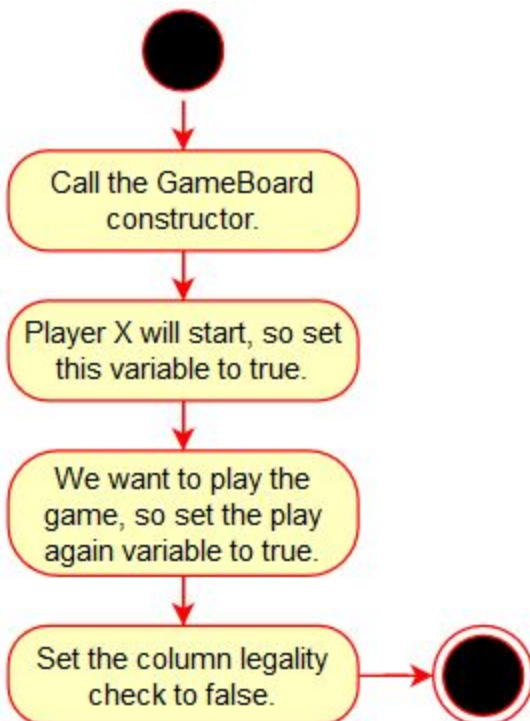
public GameBoard():



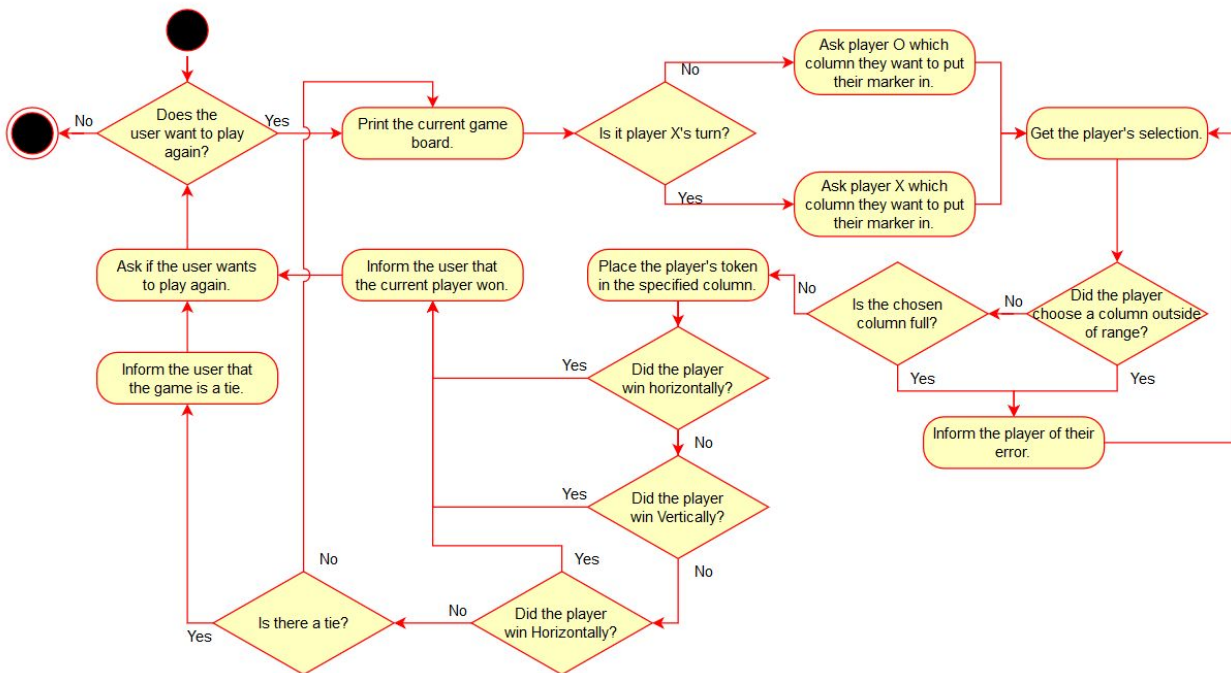
GameScreen class:



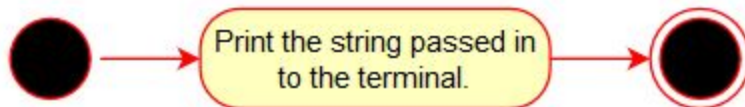
public GameScreen():



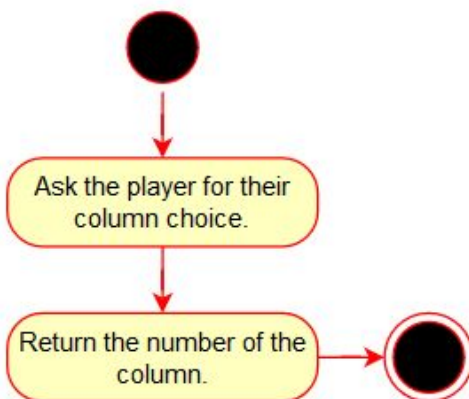
```
public static void main(String [] args):
```



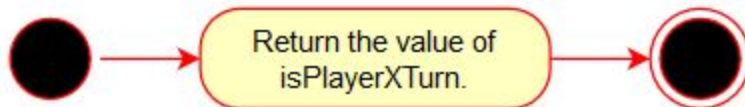
```
public void printGameBoard(string):
```



```
public int getPlayersChoice():
```



```
public bool getIsPlayerXTurn():
```



```
public void setIsPlayerXTurn(bool):
```

