FUNCTION chooseGameMode():

RETURN PROMPT("1 for human vs human, 2 for human vs computer") == 2

FUNCTION initBoard():

FOR i FROM 0 TO 2:

FOR j FROM 0 TO 2:

board[i][j] = ' '

FUNCTION gameBoard():

FOR i FROM 0 TO 2:

FOR j FROM 0 TO 2:

PRINT board[i][j] + IF j < 2 THEN "|" ELSE ""

PRINT NEWLINE + IF i < 2 THEN "-----" ELSE ""

FUNCTION yourTurn():

WHILE NOT validMove DO:

gameBoard()

move = getMove()

IF isValidMove(move):

board[move[0]][move[1]] = currentPlayer

validMove = true

currentPlayer = IF currentPlayer == 'X' THEN 'O' ELSE 'X'

FUNCTION getMove():

CREATE scanner

move = NEW ARRAY OF SIZE 2

PRINT "Player " + currentPlayer + ", enter your move (row column):"

READ move[0] FROM USER INPUT - 1

READ move[1] FROM USER INPUT - 1

RETURN move

FUNCTION isValidMove(move):

RETURN move[0] >= 0 AND move[0] < 3 AND move[1] >= 0 AND move[1] < 3 AND board[move[0]][move[1]] == ' '

FUNCTION machineTurn():

PRINT "Computer's turn:"

CREATE random number generator

WHILE NOT validMove DO:

row = random.nextInt(3)

col = random.nextInt(3)

IF isValidMove([row, col]):

board[row][col] = 'O'

validMove = true

currentPlayer = 'X'

gameBoard()

FUNCTION checkWinner():

FOR i FROM 0 TO 2:

IF board[i][0] == currentPlayer AND board[i][1] == currentPlayer AND board[i][2] == currentPlayer:

RETURN currentPlayer

FOR j FROM 0 TO 2:

IF board[0][j] == currentPlayer AND board[1][j] == currentPlayer AND board[2][j] == currentPlayer:

RETURN currentPlayer

IF (board[0][0] == currentPlayer AND board[1][1] == currentPlayer AND board[2][2] == currentPlayer) OR

(board[0][2] == currentPlayer AND board[1][1] == currentPlayer AND board[2][0] == currentPlayer):

RETURN currentPlayer

IF ALL CELLS ARE OCCUPIED:

RETURN 'D'

FUNCTION main():

humanVsComputer = chooseGameMode()

initBoard()

winner = ' '

WHILE winner == ' ' DO:

IF humanVsComputer:

IF currentPlayer == 'X':

yourTurn()

ELSE:

machineTurn()

ELSE:

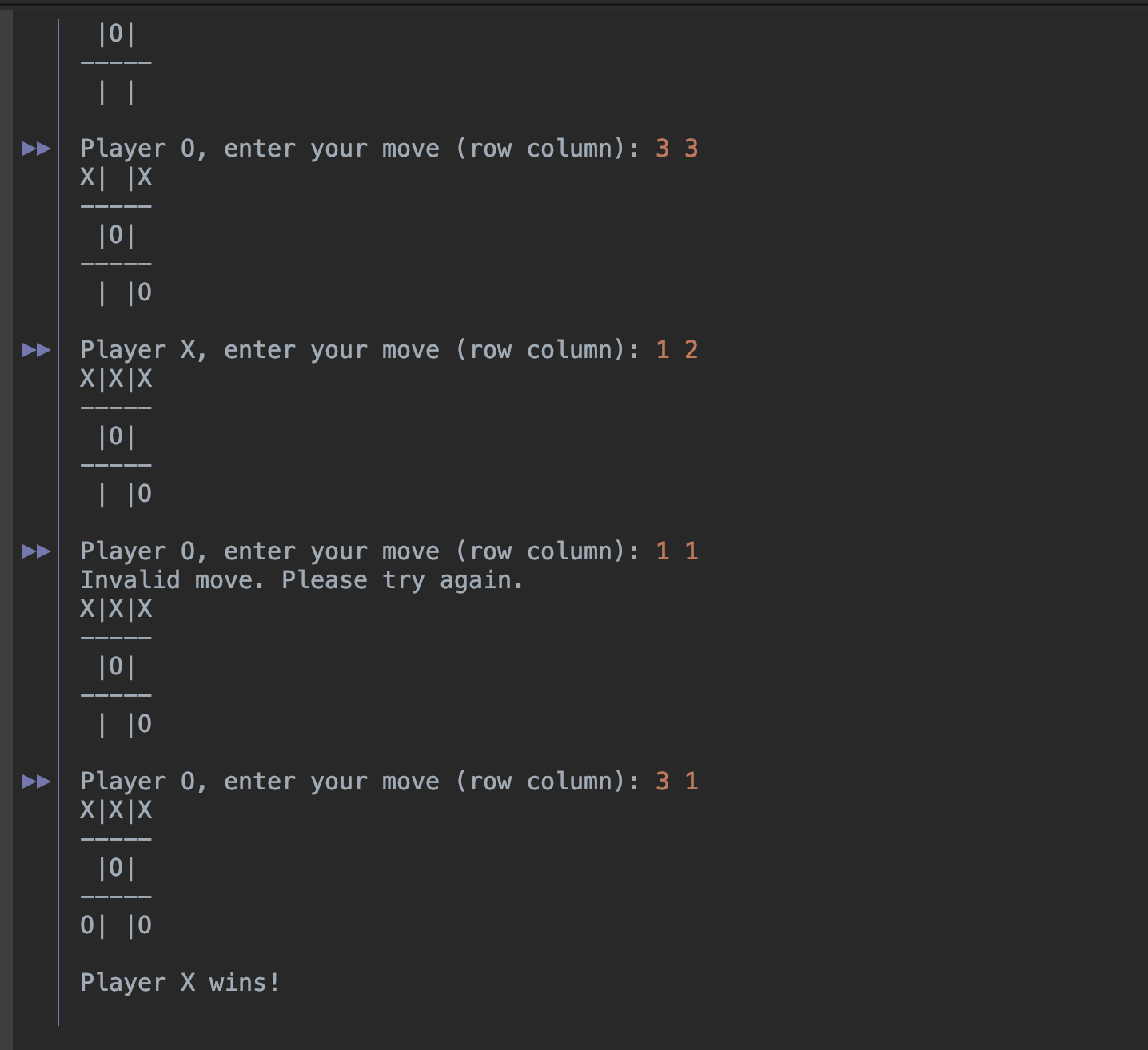
yourTurn()

winner = checkWinner()

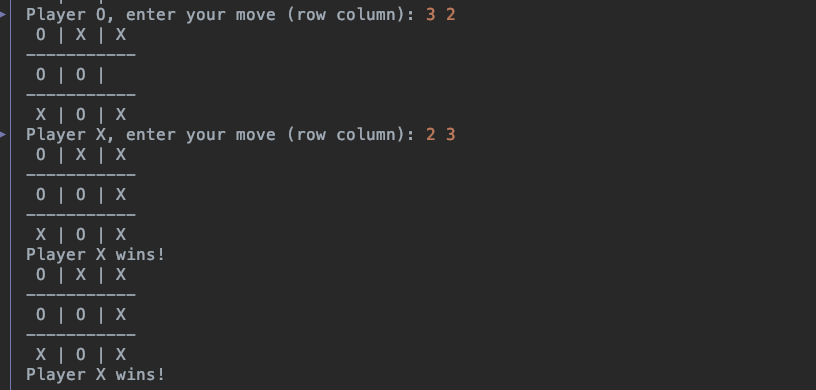
gameBoard()

PRINT IF winner == 'D' THEN "It's a draw!" ELSE "Player " + winner + " wins!"

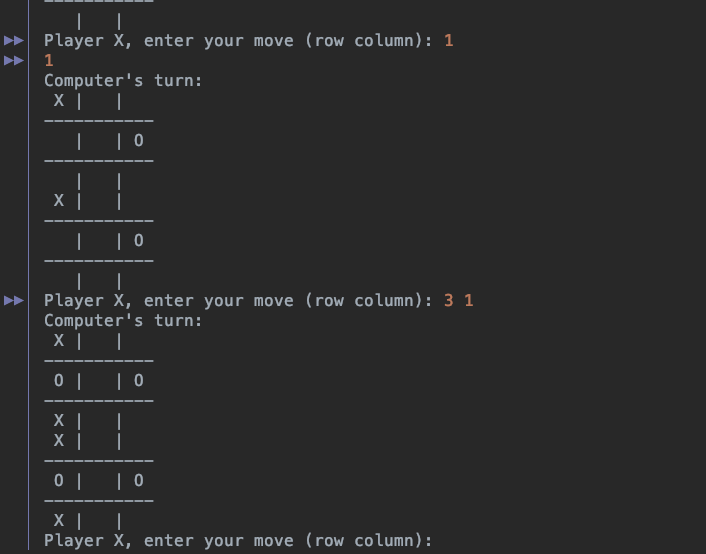
**ERRORS:**



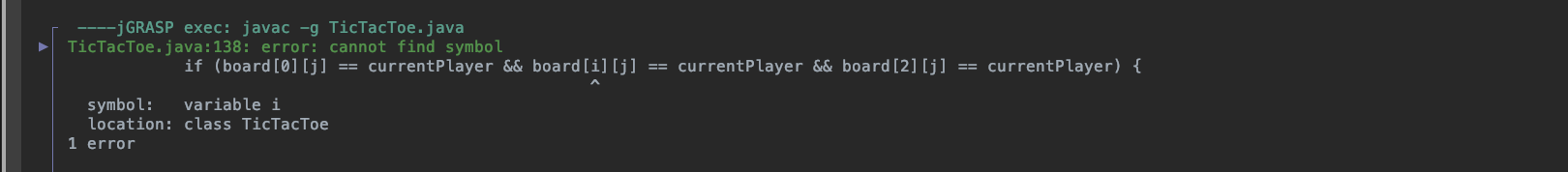
Prompting Player O for a move after player X wins. Fixed by including a break statement in YourTurn().



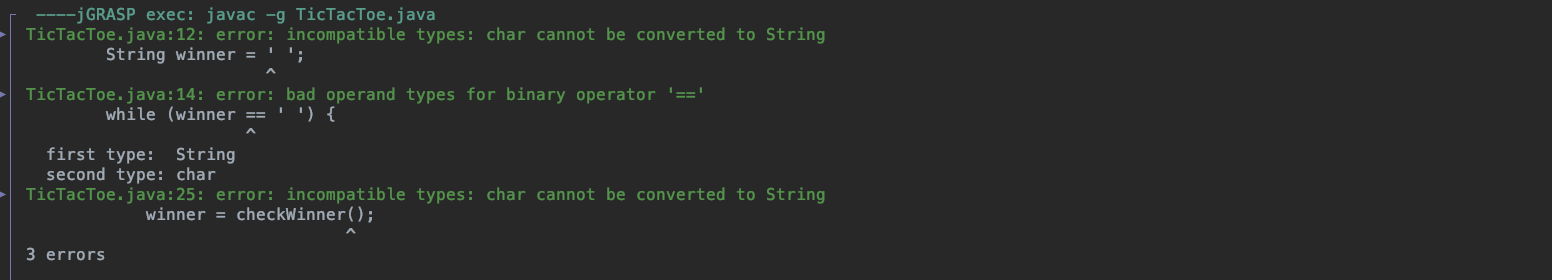
Gameboard would print twice when one player would win. Changed by removing the method call in main() and placed in yourTurn() only.



Formatting issue with printing out the gameBoard, changed the loop within the method to fix the issue.



Winning logic was wrong, fixed it by creating three separate if statements checking for the three possible ways to win.



Incorrect data type, changed it to char.

TEST RUNS:

