NAME: Célie Pierre
COS 161, Spring 2022, Prof. Andrew Amorelli
Project 01

Part 1 - Regular Tic Tac Toe

takeTurn()

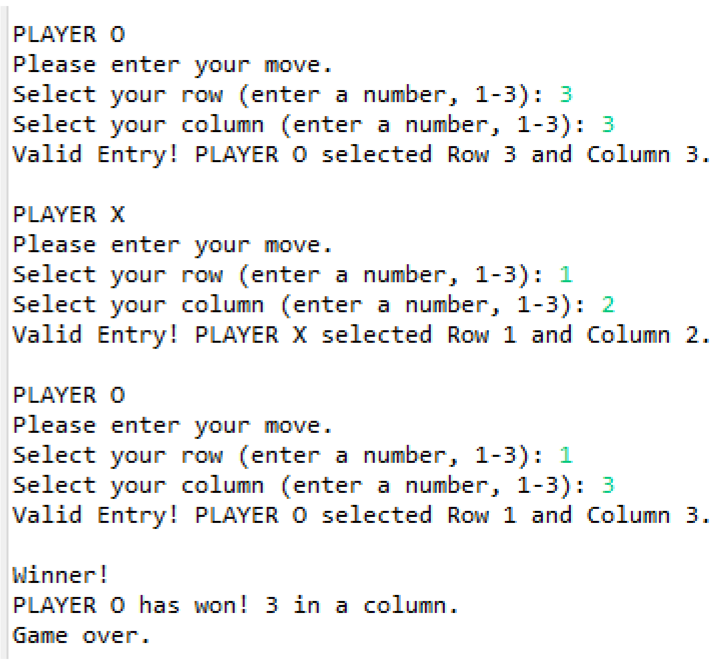
```
    public void takeTurn(char p) {

        Scanner scanner = new Scanner(System.in);

        boolean validEntry = false;
        while (!validEntry) {
            // Prompts user to enter input in the console for the coordinates of their move
            System.out.print("PLAYER " + p + "\n"
                             + "Please enter your move. \n"
                             + "Select your row (enter a number, 1-3): ");
            int row = scanner.nextInt();
            System.out.print("Select your column (enter a number, 1-3): ");
            int col = scanner.nextInt();

            if (row == 1 || row == 2 || row == 3) {
                if (col == 1 || col == 2 || col == 3) {
                    if (gameBoard[row-1][col-1] == 0) {
                        System.out.println("Valid Entry! PLAYER " + p + " selected Row "
                                           + row + " and Column " + col + ".\n");
                        gameBoard[row-1][col-1] = p;
                        drawBoard();
                        validEntry = true;
                    } else {
                        System.out.println("Invalid Entry! Move already selected. "
                                         + "Please choose again.");
                    }
                } else {
                    System.out.println("Invalid Column Entry! Please enter a "
                                     + "valid number 1-3.");
                }
            } else {
                System.out.println("Invalid Row Entry! Please enter a "
                                 + "valid number 1-3.");
            }
        }
    }
```

checkWin()

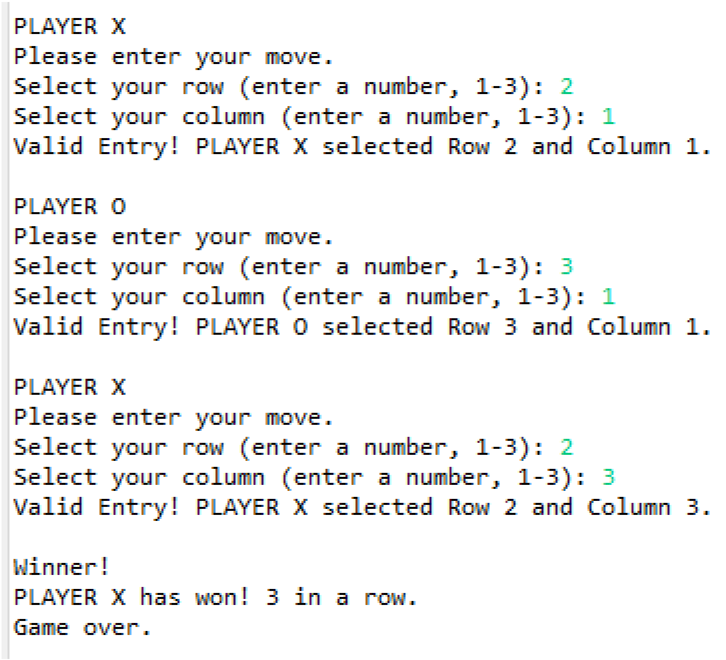
```
    public char checkWin() {
        char winner = 'N'; // No winner

        // Check for draw
        int empty = 0;
        for (int r = 0; r < 3; ++r) {
            for (int c = 0; c < 3; ++c) {
                if (gameBoard[r][c] == 0) {
                    ++empty;
                }
            }
        }
        if (empty == 0) {
            System.out.println("DRAW! Game over.");
            winner = 'D';
        }

        // Check for 3 in a row
        for (int r = 0; r < 3; ++r) {
            int c = 0;
            if (gameBoard[r][c] != 0) {
                if (gameBoard[r][c] == gameBoard[r][c+1]
                        && gameBoard[r][c] == gameBoard[r][c+2]) {
                    System.out.println("Winner!");
                    winner = gameBoard[r][c];
                    System.out.println("PLAYER " + gameBoard[r][c] + " has won! "
                                     + "3 in a row.\nGame over.");
                }
            }
        }

        // Check for 3 in a column
        for (int c = 0; c < 3; ++c) {
            int r = 0;
            if (gameBoard[r][c] != 0) {
                if (gameBoard[r][c] == gameBoard[r+1][c]
                        && gameBoard[r][c] == gameBoard[r+2][c]) {
                    System.out.println("Winner!");
                    winner = gameBoard[r][c];
                    System.out.println("PLAYER " + gameBoard[r][c] + " has won! "
                                     + "3 in a column.\nGame over.");
                }
            }
        }

        // Check for diagonal - left to right
        int r = 0;
        int c = 0;
        if (gameBoard[r][c] != 0) {
            if (gameBoard[r][c] == gameBoard[r+1][c+1]
                    && gameBoard[r][c] == gameBoard[r+2][c+2]) {
                System.out.println("Winner!");
                winner = gameBoard[r][c];
                System.out.println("PLAYER " + gameBoard[r][c] + " has won! "
                                 + "Diagonal - left to right.\nGame over.");
            }
        }

        // Check for diagonal - right to left
        r = 0;
        c = 0;
        if (gameBoard[r+2][c] != 0) {
            if (gameBoard[r+2][c] == gameBoard[r+1][c+1]
                    && gameBoard[r+2][c] == gameBoard[r][c+2]) {
                System.out.println("Winner!");
                winner = gameBoard[r][c];
                System.out.println("PLAYER " + gameBoard[r][c] + " has won! "
                                 + "Diagonal - right to left\nGame over.");
            }
        }


        return winner;
```

```
    }
```

Screenshots - Console

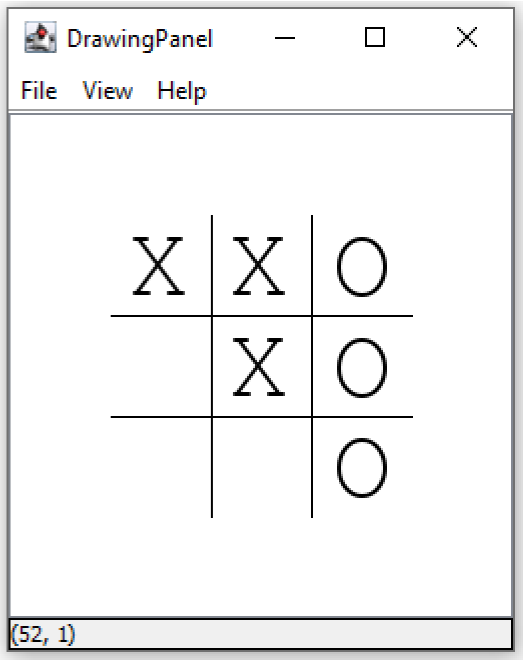


```
PLAYER O
Please enter your move.
Select your row (enter a number, 1-3): 3
Select your column (enter a number, 1-3): 3
Valid Entry! PLAYER O selected Row 3 and Column 3.

PLAYER X
Please enter your move.
Select your row (enter a number, 1-3): 1
Select your column (enter a number, 1-3): 2
Valid Entry! PLAYER X selected Row 1 and Column 2.

PLAYER O
Please enter your move.
Select your row (enter a number, 1-3): 1
Select your column (enter a number, 1-3): 3
Valid Entry! PLAYER O selected Row 1 and Column 3.

Winner!
PLAYER O has won! 3 in a column.
Game over.
```

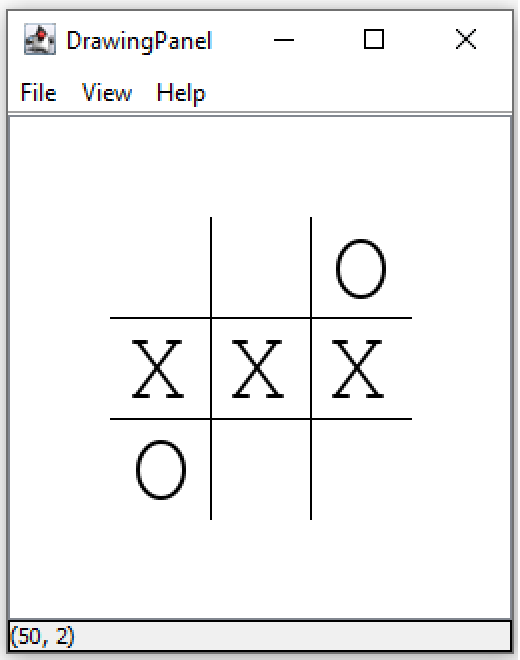


```
PLAYER X
Please enter your move.
Select your row (enter a number, 1-3): 2
Select your column (enter a number, 1-3): 1
Valid Entry! PLAYER X selected Row 2 and Column 1.

PLAYER O
Please enter your move.
Select your row (enter a number, 1-3): 3
Select your column (enter a number, 1-3): 1
Valid Entry! PLAYER O selected Row 3 and Column 1.

PLAYER X
Please enter your move.
Select your row (enter a number, 1-3): 2
Select your column (enter a number, 1-3): 3
Valid Entry! PLAYER X selected Row 2 and Column 3.

Winner!
PLAYER X has won! 3 in a row.
Game over.
```

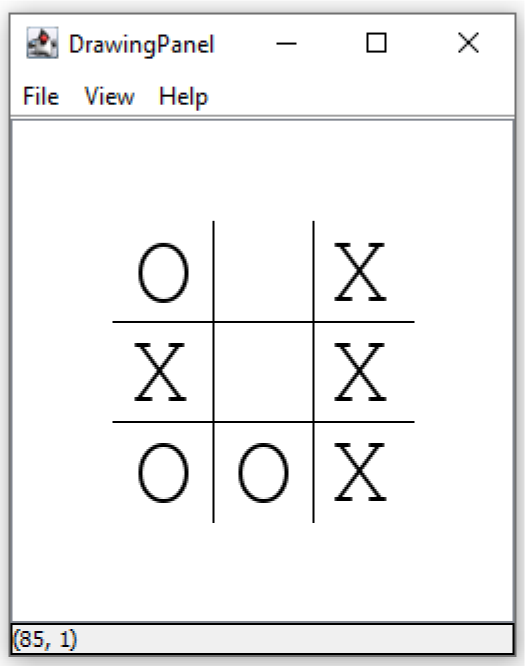
Screenshots - Drawing Panel



X	X	O
	X	O
		O



		O
X	X	X
O		



O		X
X		X
O	O	X

Part 2 - Advanced Tic Tac Toe

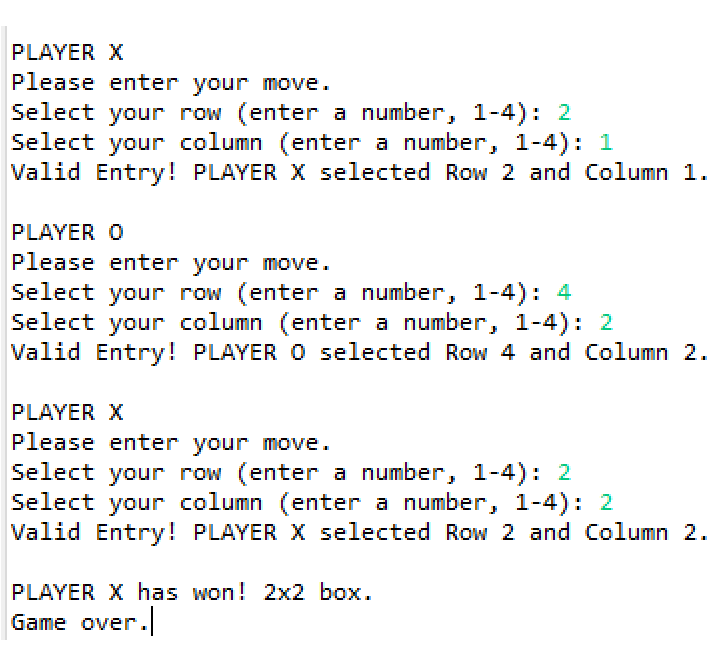
takeTurn()

```
    public void takeTurn(char p) {

        Scanner scanner = new Scanner(System.in);

        boolean validEntry = false;
        while (!validEntry) {
            // Prompts user to enter input in the console for the coordinates of their move
            System.out.print("PLAYER " + p + "\n"
                             + "Please enter your move. \n"
                             + "Select your row (enter a number, 1-" + rSize + "): ");
            int row = scanner.nextInt();
            System.out.print("Select your column (enter a number, 1-" + cSize + "): ");
            int col = scanner.nextInt();

            if (row == 1 || row == 2 || row == 3 || row == 4) {
                if (col == 1 || col == 2 || col == 3 || col == 4) {
                    if (gameBoard[row-1][col-1] == 0) {
                        System.out.println("Valid Entry! PLAYER " + p + " selected Row "
                                           + row + " and Column " + col + ".\n");
                        gameBoard[row-1][col-1] = p;
                        drawBoard();
                        validEntry = true;
                    } else {
                        System.out.println("Invalid Entry! Move already selected. "
                                         + "Please choose again.");
                    }
                } else {
                    System.out.println("Invalid Column Entry! Please enter a "
                                     + "valid number 1-" + cSize + ".");
                }
            } else {
                System.out.println("Invalid Row Entry! Please enter a "
                                 + "valid number 1-" + rSize + ".");
            }
        }
    }
```

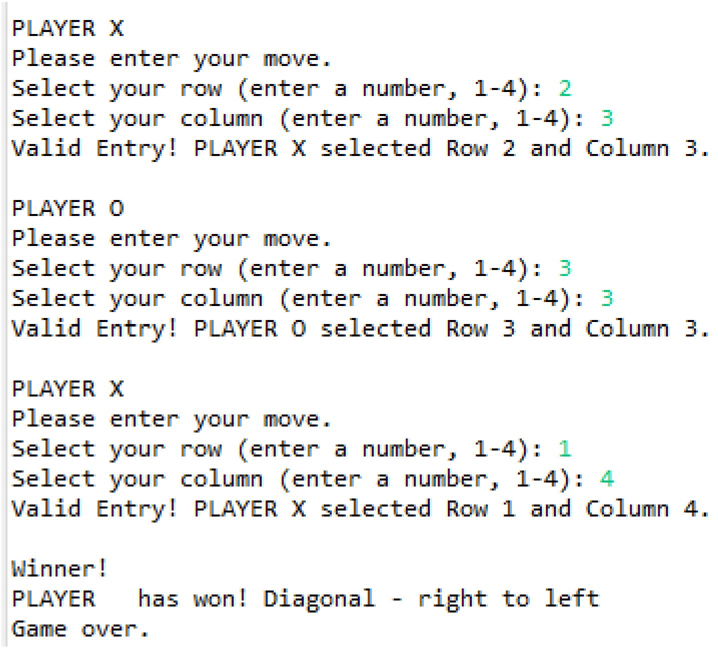
checkWin()

```
    public char checkWin() {
        char winner = 'N'; // No winner

        // Check for draw
        int empty = 0;
        for (int r = 0; r < rSize; ++r) {
            for (int c = 0; c < cSize; ++c) {
                if (gameBoard[r][c] == 0) {
                    ++empty;
                }
            }
        }
        if (empty == 0) {
            System.out.println("DRAW! Game over.");
            winner = 'D';
        }

        // Check for 4 in a row
        for (int r = 0; r < rSize; ++r) {
            int c = 0;
            if (gameBoard[r][c] != 0) {
                if (gameBoard[r][c] == gameBoard[r][c+1]
                        && gameBoard[r][c] == gameBoard[r][c+2]
                                && gameBoard[r][c] == gameBoard[r][c+3]) {
                    System.out.println("Winner!");
                    winner = gameBoard[r][c];
                    System.out.println("PLAYER " + gameBoard[r][c] + " has won! "
                                     + "3 in a row.\nGame over.");
                }
            }
        }

        // Check for 4 in a column
        for (int c = 0; c < cSize; ++c) {
            int r = 0;
            if (gameBoard[r][c] != 0) {
                if (gameBoard[r][c] == gameBoard[r+1][c]
                        && gameBoard[r][c] == gameBoard[r+2][c]
                                && gameBoard[r][c] == gameBoard[r+3][c]) {
                    System.out.println("Winner!");
                    winner = gameBoard[r][c];
                    System.out.println("PLAYER " + gameBoard[r][c] + " has won! "
                                     + "3 in a column.\nGame over.");
                }
            }
        }

        // Check for diagonal - left to right
        int r = 0;
        int c = 0;
        if (gameBoard[r][c] != 0) {
            if (gameBoard[r][c] == gameBoard[r+1][c+1]
                    && gameBoard[r][c] == gameBoard[r+2][c+2]
                            && gameBoard[r][c] == gameBoard[r+3][c+3]) {
                System.out.println("Winner!");
                winner = gameBoard[r][c];
                System.out.println("PLAYER " + gameBoard[r][c] + " has won! "
                                 + "Diagonal - left to right.\nGame over.");
            }
        }

        // Check for diagonal - right to left
        r = 0;
        c = 0;
        if (gameBoard[r+3][c] != 0) {
            if (gameBoard[r+3][c] == gameBoard[r+2][c+1]
                    && gameBoard[r+3][c] == gameBoard[r+1][c+2]
                            && gameBoard[r+3][c] == gameBoard[r][c+3]) {
                System.out.println("Winner!");
                winner = gameBoard[r][c];
                System.out.println("PLAYER " + gameBoard[r][c] + " has won! "
                                 + "Diagonal - right to left\nGame over.");
            }
        }

        // Check for 2x2 box
        for (r = 0; r < rSize-1; ++r) {
            for (c = 0; c < cSize-1; ++c) {
                if (gameBoard[r][c] != 0) {
                    if (gameBoard[r][c] == gameBoard[r+1][c]
                            && gameBoard[r][c] == gameBoard[r][c+1]
                                    && gameBoard[r][c] == gameBoard[r+1][c+1]) {
                        winner = gameBoard[r][c];
                        System.out.println("PLAYER " + gameBoard[r][c] + " has won! "
                                 + "2x2 box. \nGame over.");
                    }
                }
            }
        }


        return winner;
    }
```

Screenshots - Console

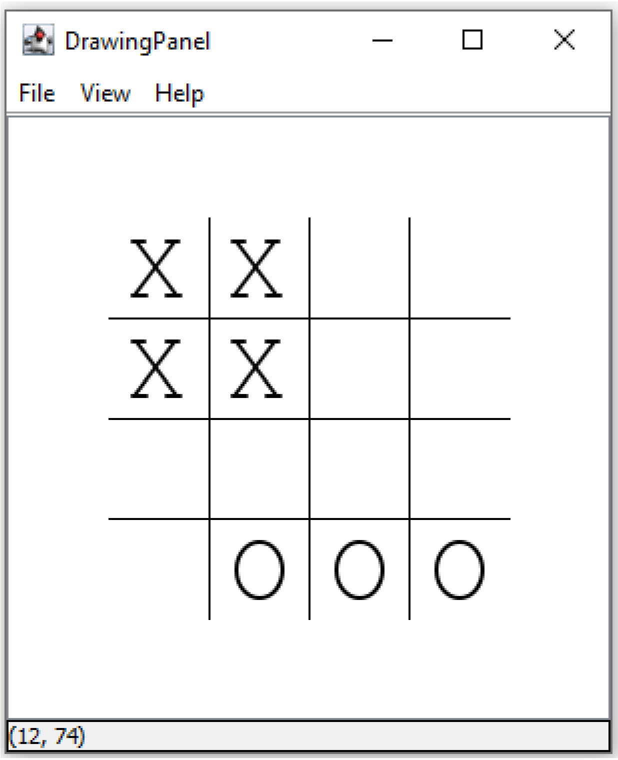


```
PLAYER X
Please enter your move.
Select your row (enter a number, 1-4): 2
Select your column (enter a number, 1-4): 1
Valid Entry! PLAYER X selected Row 2 and Column 1.

PLAYER O
Please enter your move.
Select your row (enter a number, 1-4): 4
Select your column (enter a number, 1-4): 2
Valid Entry! PLAYER O selected Row 4 and Column 2.

PLAYER X
Please enter your move.
Select your row (enter a number, 1-4): 2
Select your column (enter a number, 1-4): 2
Valid Entry! PLAYER X selected Row 2 and Column 2.

PLAYER X has won! 2x2 box.
Game over.
```

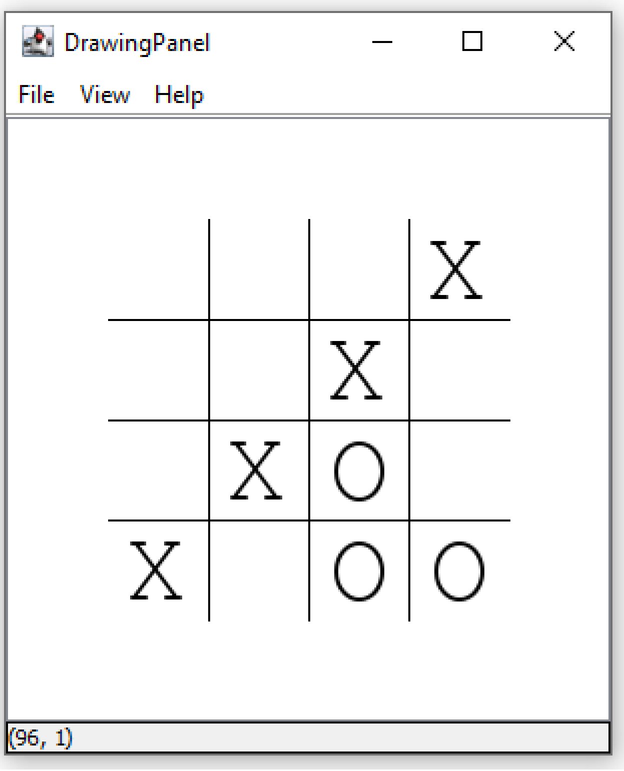


```
PLAYER X
Please enter your move.
Select your row (enter a number, 1-4): 2
Select your column (enter a number, 1-4): 3
Valid Entry! PLAYER X selected Row 2 and Column 3.

PLAYER O
Please enter your move.
Select your row (enter a number, 1-4): 3
Select your column (enter a number, 1-4): 3
Valid Entry! PLAYER O selected Row 3 and Column 3.

PLAYER X
Please enter your move.
Select your row (enter a number, 1-4): 1
Select your column (enter a number, 1-4): 4
Valid Entry! PLAYER X selected Row 1 and Column 4.

Winner!
PLAYER   has won! Diagonal - right to left
Game over.
```

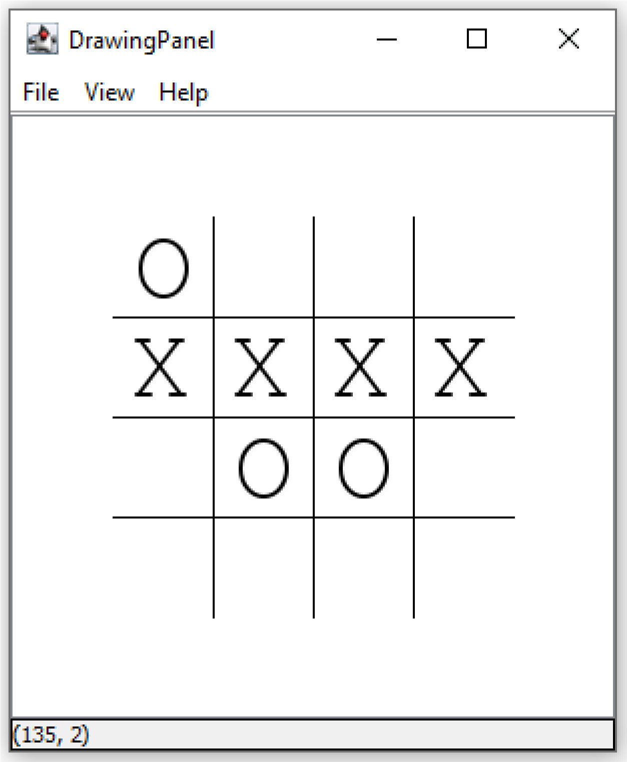
Screenshots - Drawing Panel



X	X		
X	X		
	O	O	O



			X
		X	
	X	O	
X		O	O

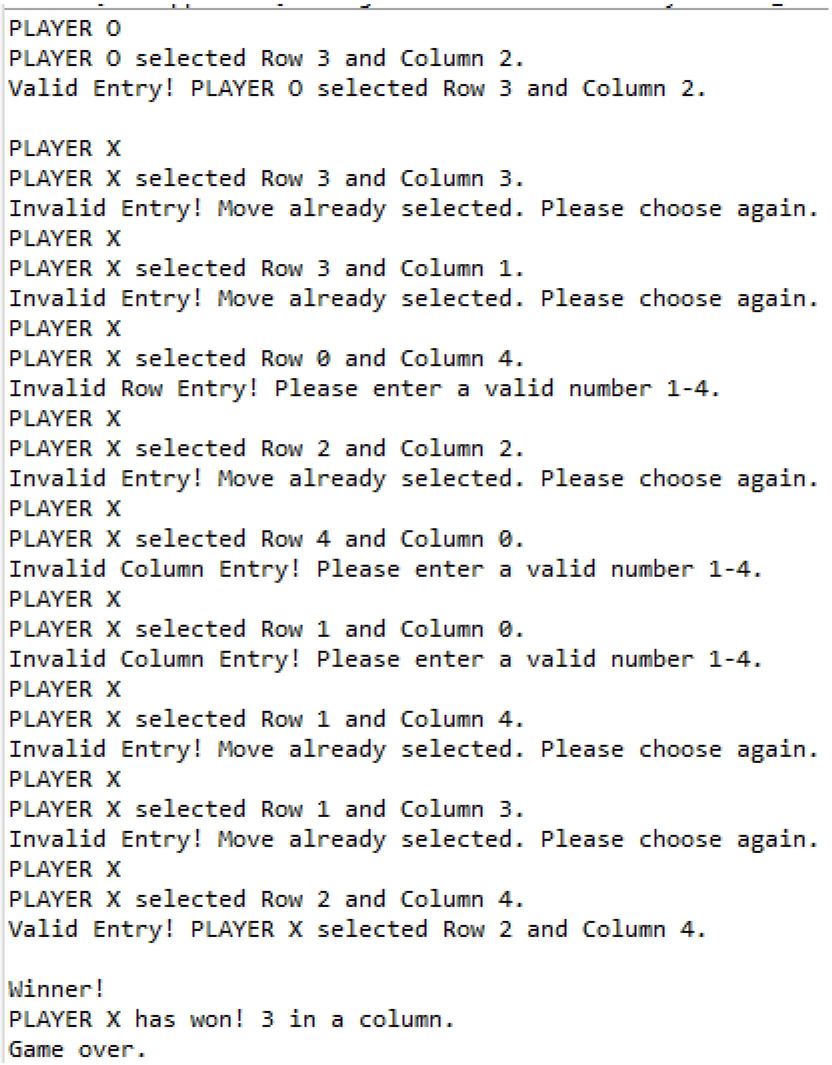


O			
X	X	X	X
	O	O	

(135, 2)

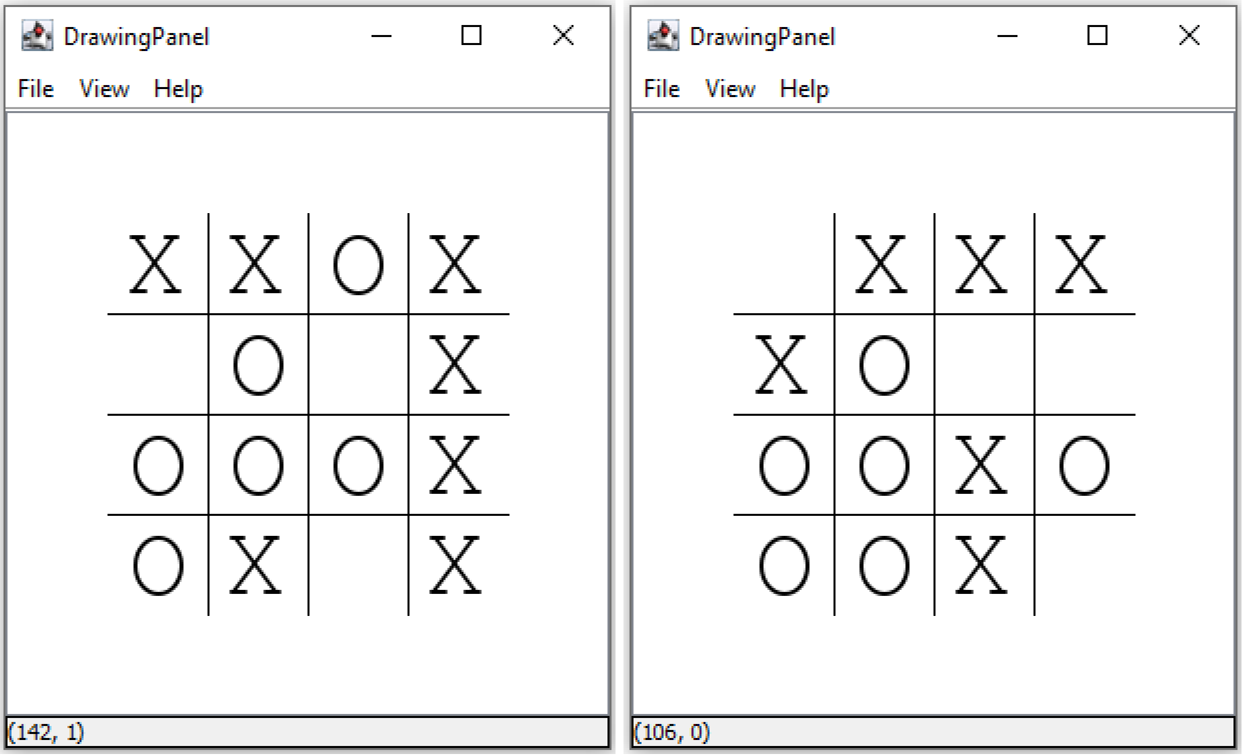
Extra Credit - Artificial Intelligence

Screenshots - Console



```
PLAYER O
PLAYER O selected Row 3 and Column 2.
Valid Entry! PLAYER O selected Row 3 and Column 2.

PLAYER X
PLAYER X selected Row 3 and Column 3.
Invalid Entry! Move already selected. Please choose again.
PLAYER X
PLAYER X selected Row 3 and Column 1.
Invalid Entry! Move already selected. Please choose again.
PLAYER X
PLAYER X selected Row 0 and Column 4.
Invalid Row Entry! Please enter a valid number 1-4.
PLAYER X
PLAYER X selected Row 2 and Column 2.
Invalid Entry! Move already selected. Please choose again.
PLAYER X
PLAYER X selected Row 4 and Column 0.
Invalid Column Entry! Please enter a valid number 1-4.
PLAYER X
PLAYER X selected Row 1 and Column 0.
Invalid Column Entry! Please enter a valid number 1-4.
PLAYER X
PLAYER X selected Row 1 and Column 4.
Invalid Entry! Move already selected. Please choose again.
PLAYER X
PLAYER X selected Row 1 and Column 3.
Invalid Entry! Move already selected. Please choose again.
PLAYER X
PLAYER X selected Row 2 and Column 4.
Valid Entry! PLAYER X selected Row 2 and Column 4.

Winner!
PLAYER X has won! 3 in a column.
Game over.
```

Screenshots - Drawing Panel



X	X	O	X
	O		X
O	O	O	X
O	X		X

(142, 1)

	X	X	X
X	O		
O	O	X	O
O	O	X	

(106, 0)