

**Ahmed Muntasir Hossain**

### **Description of the Lab**

The purpose of the lab was to design a game that would be displayed on the OLED screen. The game would be controlled using a joystick. For the purpose of this lab, I developed a Tic Tac Toe game. The screen has a grid consisting of 9 rectangles which are labeled as box1, box2, box3, box4, box5, box6, box7, box8, and box9 beginning from the top left of the screen. Additionally, there is a smaller rectangle, cursor, which defaults to box5 when the game starts. Using the joystick, the cursor can be moved to any one of the boxes. The player presses the joystick push button to input their move on the screen. Player 1 inputs the cross “X” and Player 2 inputs the circle “O”. The player who is able to place three of their marks first in a vertical, horizontal, or diagonal row is the winner. In the game, if the first player wins then the program displays “Player 1 Won”, if the second player wins it displays “Player 2 Won”, and if there is a tie it displays “Tie Game” on the screen.

The joystick is an analog device which provides a voltage input from 0 to 3.3V to the Arduino, which is then converted to a 10-bit binary number by the Analog Digital Converter (ADC). This voltage input from the joystick can be varied by the user by moving the joystick along the x and y-axis. The device and the program was set up such that if the ADC value from the VRy pin which controls the x-axis or the VRx pin which controls the y-axis on the joystick were to be greater than 400, then the cursor would move to box on its right or above it, respectively, depending on the axis being controlled. However, if the input is less than 300, then the cursor would move to the box on its left or below it, respectively. If the joystick is not moved in a particular direction on a certain axis, then the ADC value is approximately 335. Therefore, within the 300 – 400 range for the ADC value on the x and y-axis, the cursor does not move.

### **Power**

The breadboard has two columns marked + (positive) and – (negative). One wire is connected to each column and it can be placed in any hole on that column. This is used to supply power from the Arduino to the breadboard. The positive and negative wires connected to the breadboard are connected to the 3.3V power pin and GND power pin on the Arduino, respectively.

### **Output**

The output of this device is the OLED display. It is an electrical component consisting of four pins GND, VCC, SCL, and SDA which are connected parallelly on the breadboard (different rows). The GND pin is connected to the negative line, the VCC pin is connected to the positive line, the SCL pin is connected to analog input pin A5 and the SDA pin is connected to the analog input pin A4. This completes the wiring connection of the OLED display.

### **Input**

The inputs of this device are the joystick and the Universal Asynchronous Receiver-Transmitter (UART) port.

The UART port on the Arduino is referred to as COM3 on my device. The name of the port may vary depending on the Arduino. The port is connected to a PC using a USB 2.0 cable (type A/B).

**Ahmed Muntasir Hossain**

Using the Serial Monitor, a message can be sent to the Arduino to be processed. Using the same port, a message can be transmitted to the PC to be displayed on the Serial Monitor.

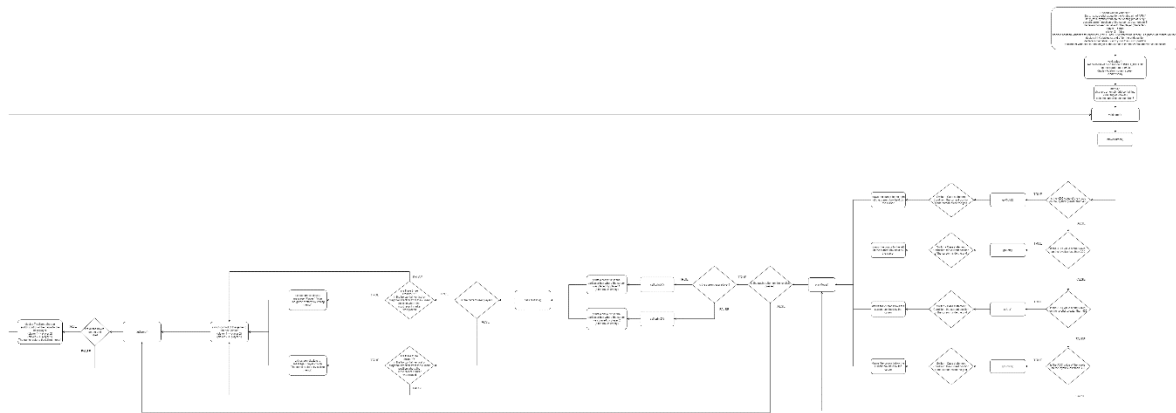
The joystick has 5 pins “GND”, “+5V”, “VRx”, “VRy”, and “SW” which are connected parallelly on the breadboard. One wire is connected to each pin serially. The +3.3V pin wire is connected to the positive line, the GND pin wire is connected to the negative line, and the “VRx” and “VRy” pin wires are connected to any two analog input pins (pin number A0 – A5) on the Arduino. The SW pin is a switch which is used to control the push button on the joystick, and it is connected to a digital i/o pin (pin number 2 – 13). This completes the wiring of the joystick.

**Parts of Arduino**

I learned about the OLED display which can be used to show different shapes, lines, images, bit maps and more. The display is connected to the Arduino using the Inter-Integrate Circuit (I2C) bus. It is a synchronous serial communication protocol which contains a single shared clock for each CPU. Additionally, I learned about the Adafruit library which consists of several functions that can be called to create a graphical user interface GUI.

Ahmed Muntasir Hossain

## Flow Chart of the Program Logic



**Please note:** Quality degraded when inserting image. Flowchart is available as a separate image in the submission folder.

## Code

```
/******
```

This is an example for our Monochrome OLEDs based on SSD1306 drivers

Pick one up today in the adafruit shop!

-----> [http://www.adafruit.com/category/63\\_98](http://www.adafruit.com/category/63_98)

This example is for a 128x32 pixel display using I2C to communicate

3 pins are required to interface (two I2C and one reset).

Adafruit invests time and resources providing this open source code, please support Adafruit and open-source hardware by purchasing products from Adafruit!

Written by Limor Fried/Ladyada for Adafruit Industries,  
with contributions from the open source community.  
BSD license, check license.txt for more information  
All text above, and the splash screen below must be  
included in any redistribution.

```
*****/
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

### Ahmed Muntasir Hossain

```
// y axis top = 679 (greater than 400)

// y axis middle = 335

// y axis bottom = 0 (less than 300)


// x axis right = 679 (greater than 400)

// x axis middle = 339

// x axis left = 0 (less than 300)


#include <SPI.h>

#include <Wire.h>

#include <Adafruit_GFX.h>

#include <Adafruit_SSD1306.h>


#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 32 // OLED display height, in pixels


const int x_axisJoystick = A0; //analog input pin for x-axis of the joystick
const int y_axisJoystick = A1; //analog input pin for y-axis of the joystick
const int BUTTON = 7; //push button


int adcValueX;

int adcValueY;


int x_1; //x coordinate of the first vertical line on the grid
int x_2; //x coordinate of the second vertical line on the grid
int y_1; //y coordinate of the first horizontal line on the grid
int y_2; //y coordinate of the second horizontal line on the grid


int xCursor;

int yCursor;

int xLength;

int yLength;


int flag = 5; //position of the cursor in the grid (default: box number 5)


bool character = true; //player 1 = true, player 2 = false
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
bool box1;  
bool box2;  
bool box3;  
bool box4;  
bool box5;  
bool box6;  
bool box7;  
bool box8;  
bool box9;
```

```
bool box1X;  
bool box2X;  
bool box3X;  
bool box4X;  
bool box5X;  
bool box6X;  
bool box7X;  
bool box8X;  
bool box9X;
```

```
bool box1O;  
bool box2O;  
bool box3O;  
bool box4O;  
bool box5O;  
bool box6O;  
bool box7O;  
bool box8O;  
bool box9O;
```

```
// Declaration for an SSD1306 display connected to I2C (SDA, SCL pins)
```

```
#define OLED_RESET 4 // Reset pin # (or -1 if sharing Arduino reset pin)
```

```
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
```

```
void setup()
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
{  
    pinMode(BUTTON, INPUT_PULLUP);           //set pinMode of push button to INPUT_PULLUP  
    Serial.begin(9600);                       //set baud rate to 9600  
  
    if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C))  
    {  
        Serial.println(F("SSD1306 allocation failed"));  
        for (;;) ;  
    }  
    display.display();  
    delay(10);  
    display.clearDisplay();  
  
    //displays menu  
    menu();  
}  
  
void loop()  
{  
    moveCursor();  
    playMove();  
    tieGame();  
}  
  
//displays the menu  
//empty grid containing 9 boxes and the cursor  
//the cursor defaults to box5 (center box) during the start of the game  
void menu()  
{  
    box1 = true;  
    box2 = true;  
    box3 = true;  
    box4 = true;  
    box5 = true;  
    box6 = true;  
    box7 = true;
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
box8 = true;
```

```
box9 = true;
```

```
box1X = false;
```

```
box2X = false;
```

```
box3X = false;
```

```
box4X = false;
```

```
box5X = false;
```

```
box6X = false;
```

```
box7X = false;
```

```
box8X = false;
```

```
box9X = false;
```

```
box1O = false;
```

```
box2O = false;
```

```
box3O = false;
```

```
box4O = false;
```

```
box5O = false;
```

```
box6O = false;
```

```
box7O = false;
```

```
box8O = false;
```

```
box9O = false;
```

```
display.setTextSize(1);
```

```
display.setTextColor(SSD1306_WHITE);
```

```
display.setCursor(30, 16);
```

```
display.println(F("Tic Tac Toe"));
```

```
display.display();
```

```
delay(3000);
```

```
display.clearDisplay();
```

```
x_1 = (display.width() - 1) / 3;
```

```
x_2 = x_1 * 2;
```

```
y_1 = (display.height() - 1) / 3;
```

```
y_2 = y_1 * 2;
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

### Ahmed Muntasir Hossain

```
display.drawLine(x_1, 0, x_1, display.height() - 1, SSD1306_WHITE);
display.drawLine(x_2, 0, x_2, display.height() - 1, SSD1306_WHITE);
display.drawLine(0, y_1, display.width() - 1, y_1, SSD1306_WHITE);
display.drawLine(0, y_2, display.width() - 1, y_2, SSD1306_WHITE);

drawBox5();
}

//depending on the ADC value read from the x and y axis of the joystick
//the program calls the appropriate directional function (goRight, goLeft, goUp, and goDown)
void moveCursor()
{
    delay(125);
    adcValueX = analogRead(y_axisJoystick);
    adcValueY = analogRead(x_axisJoystick);

    if (adcValueX > 400 && adcValueY > 300 && adcValueY < 400)
    {
        goRight();
    }
    else if (adcValueX < 300 && adcValueY > 300 && adcValueY < 400)
    {
        goLeft();
    }
    else if (adcValueY > 400 && adcValueX > 300 && adcValueX < 400)
    {
        goUp();
    }
    else if (adcValueY < 300 && adcValueX > 300 && adcValueX < 400)
    {
        goDown();
    }
}
```



## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

//moves the cursor to a particular box on the grid depending on the its current position (flag) and the direction it is moving

void goRight()

```
{  
    switch (flag)  
    {  
        case 1:  
            drawBox2();  
            break;  
        case 2:  
            drawBox3();  
            break;  
        case 3:  
            break;  
        case 4:  
            drawBox5();  
            break;  
        case 5:  
            drawBox6();  
            break;  
        case 6:  
            break;  
        case 7:  
            drawBox8();  
            break;  
        case 8:  
            drawBox9();  
            break;  
        case 9:  
            break;  
    }  
}
```

void goLeft()

```
{  
    switch (flag)  
    {
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
case 1:
    break;
case 2:
    drawBox1();
    break;
case 3:
    drawBox2();
    break;
case 4:
    break;
case 5:
    drawBox4();
    break;
case 6:
    drawBox5();
    break;
case 7:
    break;
case 8:
    drawBox7();
    break;
case 9:
    drawBox8();
    break;
}
}
```

```
void goUp()
{
    switch (flag)
    {
        case 1:
            break;
        case 2:
            break;
        case 3:
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
        break;
    case 4:
        drawBox1();
        break;
    case 5:
        drawBox2();
        break;
    case 6:
        drawBox3();
        break;
    case 7:
        drawBox4();
        break;
    case 8:
        drawBox5();
        break;
    case 9:
        drawBox6();
        break;
    }
}
```

```
void goDown()
{
    switch (flag)
    {
        case 1:
            drawBox4();
            break;
        case 2:
            drawBox5();
            break;
        case 3:
            drawBox6();
            break;
        case 4:
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
        drawBox7();
        break;
    case 5:
        drawBox8();
        break;
    case 6:
        drawBox9();
        break;
    case 7:
        break;
    case 8:
        break;
    case 9:
        break;
    }
}

//draws the cursor (smaller rectangle that the user controls) on to a particular box on the grid
void drawBox1()
{
    display.drawRect(xCursor , yCursor, xLength, yLength, SSD1306_BLACK);
    xCursor = 3;
    xLength = x_1 - 6;
    yCursor = 2;
    yLength = y_1 - 3;
    display.drawRect(xCursor , yCursor, xLength, yLength, SSD1306_WHITE);
    display.display();
    flag = 1;
}

void drawBox2()
{
    display.drawRect(xCursor , yCursor, xLength, yLength, SSD1306_BLACK);
    xCursor = x_1 + 3;
    xLength = x_1 - 6;
    yCursor = 2;
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

### Ahmed Muntasir Hossain

```
yLength = y_1 - 3;

display.drawRect(xCursor , yCursor, xLength, yLength, SSD1306_WHITE);

display.display();

flag = 2;
}

void drawBox3()
{
    display.drawRect(xCursor , yCursor, xLength, yLength, SSD1306_BLACK);

    xCursor = x_2 + 3;

    xLength = x_1 - 6;

    yCursor = 2;

    yLength = y_1 - 3;

    display.drawRect(xCursor , yCursor, xLength, yLength, SSD1306_WHITE);

    display.display();

    flag = 3;
}

void drawBox4()
{
    display.drawRect(xCursor , yCursor, xLength, yLength, SSD1306_BLACK);

    xCursor = 3;

    xLength = x_1 - 6;

    yCursor = y_1 + 2;

    yLength = y_1 - 3;

    display.drawRect(xCursor , yCursor, xLength, yLength, SSD1306_WHITE);

    display.display();

    flag = 4;
}

void drawBox5()
{
    display.drawRect(xCursor , yCursor, xLength, yLength, SSD1306_BLACK);

    xCursor = x_1 + 3;

    xLength = x_1 - 6;

    yCursor = y_1 + 2;
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

### Ahmed Muntasir Hossain

```
yLength = y_1 - 3;
display.drawRect(xCursor, yCursor, xLength, yLength, SSD1306_WHITE);
display.display();
flag = 5;
}

void drawBox6()
{
    display.drawRect(xCursor, yCursor, xLength, yLength, SSD1306_BLACK);
    xCursor = x_2 + 3;
    xLength = x_1 - 6;
    yCursor = y_1 + 2;
    yLength = y_1 - 3;
    display.drawRect(xCursor, yCursor, xLength, yLength, SSD1306_WHITE);
    display.display();
    flag = 6;
}

void drawBox7()
{
    display.drawRect(xCursor, yCursor, xLength, yLength, SSD1306_BLACK);
    xCursor = 3;
    xLength = x_1 - 6;
    yCursor = y_2 + 2;
    yLength = y_1 - 3;
    display.drawRect(xCursor, yCursor, xLength, yLength, SSD1306_WHITE);
    display.display();
    flag = 7;
}

void drawBox8()
{
    display.drawRect(xCursor, yCursor, xLength, yLength, SSD1306_BLACK);
    xCursor = x_1 + 3;
    xLength = x_1 - 6;
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

### Ahmed Muntasir Hossain

```
yCursor = y_2 + 2;
yLength = y_1 - 3;
display.drawRect(xCursor, yCursor, xLength, yLength, SSD1306_WHITE);
display.display();
flag = 8;
}

void drawBox9()
{
    display.drawRect(xCursor, yCursor, xLength, yLength, SSD1306_BLACK);
    xCursor = x_2 + 3;
    xLength = x_1 - 6;
    yCursor = y_2 + 2;
    yLength = y_1 - 3;
    display.drawRect(xCursor, yCursor, xLength, yLength, SSD1306_WHITE);
    display.display();
    flag = 9;
}

//calls the print function for a particular player, if they click on the push button on the joystick
void playMove()
{
    if (!digitalRead(BUTTON)) //checks to see if the push button on the joystick was pressed
    {
        if (character)
        {
            printX(); //if player 1 clicks on the push button, the program prints "X"
        }
        else
        {
            printO(); //if player 2 clicks on the push button, the program prints "O"
        }
    }
}

//uses the flag to determine the box the player clicked the push button on
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

### Ahmed Muntasir Hossain

```
//calls the respective printSymbolInBox() function
```

```
String symbolX = "X";
```

```
String symbolO = "O"; //the symbol appears as ||
```

```
void printX()
```

```
{
```

```
    switch (flag)
```

```
    {
```

```
        case 1:
```

```
            if (box1)
```

```
                printSymbolInBox1(symbolX);
```

```
            break;
```

```
        case 2:
```

```
            if (box2)
```

```
                printSymbolInBox2(symbolX);
```

```
            break;
```

```
        case 3:
```

```
            if (box3)
```

```
                printSymbolInBox3(symbolX);
```

```
            break;
```

```
        case 4:
```

```
            if (box4)
```

```
                printSymbolInBox4(symbolX);
```

```
            break;
```

```
        case 5:
```

```
            if (box5)
```

```
                printSymbolInBox5(symbolX);
```

```
            break;
```

```
        case 6:
```

```
            if (box6)
```

```
                printSymbolInBox6(symbolX);
```

```
            break;
```

```
        case 7:
```

```
            if (box7)
```

```
                printSymbolInBox7(symbolX);
```

```
            break;
```



## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
case 8:
    if (box8)
        printSymbolInBox8(symbolX);
    break;
case 9:
    if (box9)
        printSymbolInBox9(symbolX);
    break;
}
}

void printO()
{
    switch (flag)
    {
        case 1:
            if (box1)
                printSymbolInBox1(symbolO);
            break;
        case 2:
            if (box2)
                printSymbolInBox2(symbolO);
            break;
        case 3:
            if (box3)
                printSymbolInBox3(symbolO);
            break;
        case 4:
            if (box4)
                printSymbolInBox4(symbolO);
            break;
        case 5:
            if (box5)
                printSymbolInBox5(symbolO);
            break;
        case 6:
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
        if (box6)
            printSymbolInBox6(symbolO);
        break;
    case 7:
        if (box7)
            printSymbolInBox7(symbolO);
        break;
    case 8:
        if (box8)
            printSymbolInBox8(symbolO);
        break;
    case 9:
        if (box9)
            printSymbolInBox9(symbolO);
        break;
    }
}

//prints the symbol inputted by the player in a particular box
void printSymbolInBox1(String symbol)
{
    display.setTextSize(1);
    display.setTextColor(SSD1306_WHITE);
    display.setCursor(19, 2);
    display.println(symbol);
    display.display();

    box1 = false;
    drawBox5();
    testBox1();
    character = !character;
}

void printSymbolInBox2(String symbol)
{
    display.setTextSize(1);
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

### Ahmed Muntasir Hossain

```
display.setTextColor(SSD1306_WHITE);  
display.setCursor(x_1 + 19, 2);  
display.println(symbol);  
display.display();
```

```
box2 = false;  
drawBox5();  
testBox2();  
character = !character;  
}
```

```
void printSymbolInBox3(String symbol)  
{  
    display.setTextSize(1);  
    display.setTextColor(SSD1306_WHITE);  
    display.setCursor(x_2 + 19, 2);  
    display.println(symbol);  
    display.display();
```

```
box3 = false;  
drawBox5();  
testBox3();  
character = !character;  
}
```

```
void printSymbolInBox4(String symbol)  
{  
    display.setTextSize(1);  
    display.setTextColor(SSD1306_WHITE);  
    display.setCursor(19, y_1 + 2);  
    display.println(symbol);  
    display.display();
```

```
box4 = false;  
drawBox5();  
testBox4();  
character = !character;
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
}  
  
void printSymbolInBox5(String symbol)  
{  
    display.setTextSize(1);  
    display.setTextColor(SSD1306_WHITE);  
    display.setCursor(x_1 + 19, y_1 + 2);  
    display.println(symbol);  
    display.display();  
  
    box5 = false;  
    drawBox5();  
    testBox5();  
    character = !character;  
}  
  
void printSymbolInBox6(String symbol)  
{  
    display.setTextSize(1);  
    display.setTextColor(SSD1306_WHITE);  
    display.setCursor(x_2 + 19, y_1 + 2);  
    display.println(symbol);  
    display.display();  
  
    box6 = false;  
    drawBox5();  
    testBox6();  
    character = !character;  
}  
  
void printSymbolInBox7(String symbol)  
{  
    display.setTextSize(1);  
    display.setTextColor(SSD1306_WHITE);  
    display.setCursor(19, y_2 + 2);  
    display.println(symbol);  
    display.display();  
  
    box7 = false;
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

### Ahmed Muntasir Hossain

```
drawBox5();

testBox7();

character = !character;
}

void printSymbolInBox8(String symbol)
{
    display.setTextSize(1);
    display.setTextColor(SSD1306_WHITE);
    display.setCursor(x_1 + 19, y_2 + 2);
    display.println(symbol);
    display.display();

    box8 = false;
    drawBox5();
    testBox8();
    character = !character;
}

void printSymbolInBox9(String symbol)
{
    display.setTextSize(1);
    display.setTextColor(SSD1306_WHITE);
    display.setCursor(x_2 + 19, y_2 + 2);
    display.println(symbol);
    display.display();

    box9 = false;
    drawBox5();
    testBox9();
    character = !character;
}

//tests if a player has three of their symbols in a row for the current box that they inputted their symbol in
void testBox1()
{
    if (character)
    {
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
    box1X = true;

    if (box1X && ((box2X && box3X) || (box4X && box7X) || (box5X && box9X)))

        player1ExitScreen();
    }

else

{

    box1O = true;

    if (box1O && ((box2O && box3O) || (box4O && box7O) || (box5O && box9O)))

        player2ExitScreen();
    }
}

void testBox2()

{

    if (character)

    {

        box2X = true;

        if (box2X && ((box1X && box3X) || (box5X && box8X)))

            player1ExitScreen();
        }

    else

    {

        box2O = true;

        if (box2O && ((box1O && box3O) || (box5O && box8O)))

            player2ExitScreen();
        }

    }

}

void testBox3()

{

    if (character)

    {

        box3X = true;

        if (box3X && ((box1X && box2X) || (box6X && box9X) || (box5X && box7X)))

            player1ExitScreen();
    }
}
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

Ahmed Muntasir Hossain

```
}  
else  
{  
    box3O = true;  
    if (box3O && ((box1O && box2O) || (box6O && box9O) || (box5O && box7O)))  
        player2ExitScreen();  
}  
  
}  
  
void testBox4()  
{  
    if (character)  
    {  
        box4X = true;  
        if (box4X && ((box1X && box7X) || (box5X && box6X)))  
            player1ExitScreen();  
    }  
    else  
    {  
        box4O = true;  
        if (box4O && ((box1O && box7O) || (box5O && box6O)))  
            player2ExitScreen();  
    }  
}  
  
void testBox5()  
{  
    if (character)  
    {  
        box5X = true;  
        if (box5X && ((box1X && box9X) || (box3X && box7X) || (box2X && box8X) || (box4X && box6X)))  
            player1ExitScreen();  
    }  
    else  
    {
```

## Game Lab 6

Course: CSCI 3331  
Section: 01

**Ahmed Muntasir Hossain**

```
    box5O = true;

    if (box5O && ((box1O && box9O) || (box3O && box7O) || (box2O && box8O) || (box4O && box6O)))

        player2ExitScreen();
    }
}

void testBox6()
{
    if (character)
    {
        box6X = true;

        if (box6X && ((box3X && box9X) || (box4X && box5X)))

            player1ExitScreen();
        }
    else
    {
        box6O = true;

        if (box6O && ((box3O && box9O) || (box4O && box5O)))

            player2ExitScreen();
        }
    }
}

void testBox7()
{
    if (character)
    {
        box7X = true;

        if (box7X && ((box1X && box4X) || (box8X && box9X) || (box3X && box5X)))

            player1ExitScreen();
        }
    else
    {
        box7O = true;

        if (box7O && ((box1O && box4O) || (box8O && box9O) || (box3O && box5O)))

            player2ExitScreen();
        }
    }
}
```



## Game Lab 6

Course: CSCI 3331  
Section: 01

Ahmed Muntasir Hossain

```
}

void testBox8()
{
    if (character)
    {
        box8X = true;

        if (box8X && ((box2X && box5X) || (box7X && box9X)))
            player1ExitScreen();
    }
    else
    {
        box8O = true;

        if (box8O && ((box2O && box5O) || (box7O && box9O)))
            player2ExitScreen();
    }
}

void testBox9()
{
    if (character)
    {
        box9X = true;

        if (box9X && ((box3X && box6X) || (box7X && box8X) || (box1X && box5X)))
            player1ExitScreen();
    }
    else
    {
        box9O = true;

        if (box9O && ((box3O && box6O) || (box7O && box8O) || (box1O && box5O)))
            player2ExitScreen();
    }
}
```

//displays the exit screen for different scenarios: Player 1 Winning, Player 2 Winning, and Tie Game

## Game Lab 6

Course: CSCI 3331  
Section: 01

### Ahmed Muntasir Hossain

```
//restarts the game
void player1ExitScreen()
{
    delay(500);
    display.clearDisplay();
    display.setTextSize(1);
    display.setTextColor(SSD1306_WHITE);
    display.setCursor(30, 16);
    display.println(F("Player 1 Won"));
    display.display();
    delay(3000);
    display.clearDisplay();
    menu();
}

void player2ExitScreen()
{
    delay(500);
    display.clearDisplay();
    display.setTextSize(1);
    display.setTextColor(SSD1306_WHITE);
    display.setCursor(30, 16);
    display.println(F("Player 2 Won"));
    display.display();
    delay(3000);
    display.clearDisplay();
    menu();
}

void tieGame()
{
    if (!box1 && !box2 && !box3 && !box4 && !box5 && !box6 && !box7 && !box8 && !box9)
    {
        delay(500);
        display.clearDisplay();
        display.setTextSize(1);
```

## **Game Lab 6**

**Course: CSCI 3331  
Section: 01**

**Ahmed Muntasir Hossain**

```
display.setTextColor(SSD1306_WHITE);  
display.setCursor(34, 16);  
display.println(F("Tie Game"));  
display.display();  
delay(3000);  
display.clearDisplay();  
character = !character;  
menu();  
}  
}
```