#include <LiquidCrystal.h> // Include the LiquidCrystal library for the LCD

// Define the pin connected to the pushbutton

const int buttonPin = 3;

// Define the LCD pin connections: RS, Enable, Data4, Data5, Data6, Data7

const int rs = 7, en = 8, d4 = 9, d5 = 10, d6 = 11, d7 = 12;

LiquidCrystal lcd(rs, en, d4, d5, d6, d7); // Create an LCD object

void setup() {

pinMode(buttonPin, INPUT); // Set the button pin as ???

lcd.begin(16, 2); // Initialize LCD type (x columns, y rows)

Serial.begin(9600); // Start Serial Monitor for debugging output

randomSeed(analogRead(0)); // Seed the random number generator

}

void loop() {

lcd.clear(); // Clear the LCD screen

lcd.print("Press to Start"); // Display instruction to the player

// Just waiting here until the player presses the button (HIGH to LOW)

while (digitalRead(buttonPin) == HIGH) {} // DONT WORRY HOW THIS WORKS NOW

// Small delay to remove mechanical button bounce noise

delay(750);

// Call the function that runs the reaction timer game

reactionGame(); // Need to be defined!!!

}

void reactionGame() {

lcd.clear(); // Clear the LCD screen

lcd.print("Get Ready..."); // Inform the player to get ready

delay(1000);

// Generate a random delay between 3 to 6 seconds (3000-6000 ms) ...

// Use the function random(); to help achieve that. Look it up how it works.

// !!! ENTER THE CODE HERE !!!

int number(random(1000,3000));

Serial.println(number);

delay(number);

lcd.clear();

lcd.clear();

lcd.print("Press NOW!"); // Ask the player to press the button as fast as possible

unsigned long startTime = millis(); // Record the time right button press

// Don’t worry about how this works, only that the code below waits until ...

// the player presses the button.

while (digitalRead(buttonPin) == HIGH) {

// This keeps running until the player reacts and presses the button

}

// Use the start time recorded above (startTime) and the current time to ...

// calculate how long the player took to press the button.

unsigned long reactionTime = millis() - startTime;

// Display the reaction time on the LCD

lcd.clear();

lcd.print("Your time:");

lcd.setCursor(0, 1); // Move to the second row

lcd.print(reactionTime); // Show the reaction time in milliseconds

lcd.print(" ms");

delay(3000); // Keep the result displayed for 3 seconds before restarting

}