

```

#define LEDPIN 13
#define SENSORPIN1 4 // First sensor
#define SENSORPIN2 5 // Second sensor

int sensorState1 = 0, lastState1 = 0;
int sensorState2 = 0, lastState2 = 0;
int countEntries = 0;
int countExits = 0;

void setup() {
    pinMode(LEDPIN, OUTPUT);
    pinMode(SENSORPIN1, INPUT);
    pinMode(SENSORPIN2, INPUT);

    digitalWrite(SENSORPIN1, HIGH); // turn on the pullup
    digitalWrite(SENSORPIN2, HIGH); // turn on the pullup

    Serial.begin(9600);
}

void loop(){
    // Read the state of both sensors
    sensorState1 = digitalRead(SENSORPIN1);
    sensorState2 = digitalRead(SENSORPIN2);

    // Check for entries and exits based on the order of sensor breaks
    if (sensorState1 == LOW && lastState1 == HIGH && sensorState2 == HIGH) {
        // Sensor 1 broken first, possible entry
        unsigned long start = millis();
        while (millis() - start < 500) { // Short delay to confirm second sensor
            sensorState2 = digitalRead(SENSORPIN2);
            if (sensorState2 == LOW) {
                countEntries++;
                Serial.print("Entries counted: ");
                Serial.println(countEntries);
                digitalWrite(LEDPIN, HIGH); // Turn LED on for entry
                delay(500); // Brief indication delay
                break;
            }
        }
    }
    else if (sensorState2 == LOW && lastState2 == HIGH && sensorState1 == HIGH) {
        // Sensor 2 broken first, possible exit
        unsigned long start = millis();
    }
}

```

```
while (millis() - start < 500) {  
  sensorState1 = digitalRead(SENSORPIN1);  
  if (sensorState1 == LOW) {  
    countExits++;  
    Serial.print("Exits detected: ");  
    Serial.println(countExits);  
    digitalWrite(LEDPIN, LOW); // Turn LED off for exit  
    delay(500);  
    break;  
  }  
}  
}  
  
// Update last states  
lastState1 = sensorState1;  
lastState2 = sensorState2;  
}
```

```
#include <SD.h> // Include the SD library

#define LEDPIN 13

#define SENSORPIN1 4 // First sensor
#define SENSORPIN2 5 // Second sensor

File logFile; // File variable to store log file

int sensorState1 = 0, lastState1 = 0;
int sensorState2 = 0, lastState2 = 0;
int countEntries = 0;
int countExits = 0;

void setup() {
    pinMode(LEDPIN, OUTPUT);
    pinMode(SENSORPIN1, INPUT);
    pinMode(SENSORPIN2, INPUT);

    digitalWrite(SENSORPIN1, HIGH); // turn on the pullup
    digitalWrite(SENSORPIN2, HIGH); // turn on the pullup

    Serial.begin(9600);
```

```

// Initialize SD card

if (!SD.begin(2)) { // Here, 2 is the CS pin number

    Serial.println("SD Card initialization failed!");

    return;

}

Serial.println("SD Card initialized.");


// Open the file, create it if it doesn't exist
logFile = SD.open("log.txt", FILE_WRITE);

if (!logFile) {

    Serial.println("Error opening log file!");

    return;

}

// If the file opened okay, write to it:
logFile.println("Starting logging...");

logFile.close();

}


void loop(){

    // Read the state of both sensors

    sensorState1 = digitalRead(SENSORPIN1);

    sensorState2 = digitalRead(SENSORPIN2);


    // Check for entries and exits based on the order of sensor breaks

```

```

if (sensorState1 == LOW && lastState1 == HIGH && sensorState2 == HIGH) {

    unsigned long start = millis();

    while (millis() - start < 500) { // Short delay to confirm second sensor

        sensorState2 = digitalRead(SENSORPIN2);

        if (sensorState2 == LOW) {

            countEntries++;

            Serial.print("Entries counted: ");

            Serial.println(countEntries);

            digitalWrite(LEDPIN, HIGH); // Turn LED on for entry

            delay(500); // Brief indication delay

            // Log entry to file

            logFile = SD.open("log.txt", FILE_WRITE);

            if (logFile) {

                logFile.println("Entry: " + String(countEntries));

                logFile.close();

            }

            break;

        }

    }

}

} else if (sensorState2 == LOW && lastState2 == HIGH && sensorState1 == HIGH) {

    unsigned long start = millis();

    while (millis() - start < 500) {

        sensorState1 = digitalRead(SENSORPIN1);

        if (sensorState1 == LOW) {

            countExits++;

        }

    }

}

```

```
Serial.print("Exits detected: ");

Serial.println(countExits);

digitalWrite(LEDPIN, LOW); // Turn LED off for exit

delay(500);

// Log exit to file

logFile = SD.open("log.txt", FILE_WRITE);

if (logFile) {

    logFile.println("Exit: " + String(countExits));

    logFile.close();

}

break;

}

}

}

// Update last states

lastState1 = sensorState1;

lastState2 = sensorState2;

}
```