2 Videos of my project (demonstration):
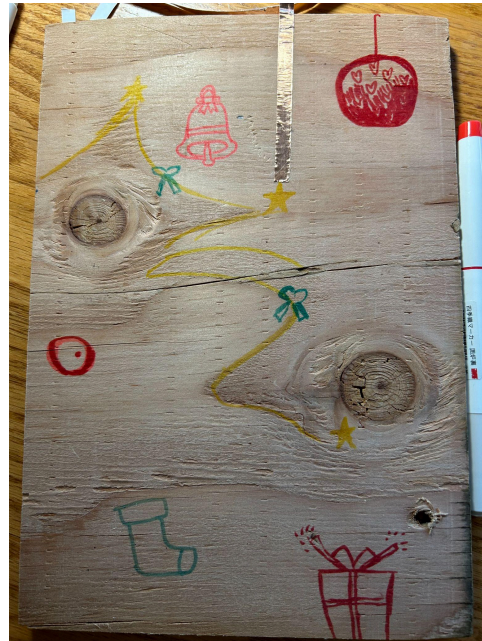https://drive.google.com/file/d/1KFbgBbQEzrAuEfUPmDOhSdhWIdQC0yzP/view?usp=sharing
https://drive.google.com/file/d/1WKm4WKkv6NsKEaDWyVyF2AaH-2XDiZt5/view?usp=sharing

Wood Door   Wood Door (with decoration)

I thought it would be cuter if I made a Christmas scene:)

The problem I had during demonstration:
https://drive.google.com/file/d/1dz-i3UqAAbKVTrqF8w_pRfPIBGMo5DLJ/view?usp=sharing
        Explanation: since I was using copper tap and use the clips to connect with circuit playground, but it did not work when I used the battery unless I plugged in to my computer

First I had a **sample code** provided by professor:

```
#include <Adafruit_CircuitPlayground.h>

int numNotes = 27;
int STmelody[] = {
```

```
  262, 294, 349, 349, 349, 349, 349, 349, 262, 294, 349, 349, 349, 349,
349, 349, 262, 294, 349, 349, 349, 349, 349, 349, 349, 349, 330
};
int STnoteDurations[] = {
  188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188,
188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188
};

int THRESHOLD = 1000;

void setup() {
  Serial.begin(9600);
  CircuitPlayground.begin();
  // led code here
}

void loop() {
  // play a melody if there is a touch, as an example
  // we'll pick pin 3, which is labeled SCL/A4 on the board
  // if the value is large enough
  if (CircuitPlayground.readCap(3) > THRESHOLD){
    // play the melody (only if the slide switch is on)
    playMelody();
  }
  else
  {
//    digitalWrite(.., LOW)
  }
  delay(50);
}

// Play a note of the specified frequency and for the specified duration.
// No tones will play if the slide switch is in the -/off position.
void playNote(int frequency, int duration) {
  // Check if the slide switch is off
  if (!CircuitPlayground.slideSwitch()) {
    // stop immediately without playing anything.
    return;
  } else {
    // Play the note for the specified duration.
```

```
      CircuitPlayground.playTone(frequency, duration, false);
      // wait for the note to play
      delay(duration + duration / 16);
  }
}


//plays stranger things theme
//collaboration with TA Claudia
void playMelody() {
  for (int i = 0; i < numNotes; i++) {
    playNote(STmelody[i], STnoteDurations[i]);
  }
}
```

Then I wanted to add my own ideas to my doorbell:

My idea is to create different touch points that are connected to different pins on the circuit board and make different colors of lights to turn on. That way, the owner of the house will know who's coming to visit and they can ask their friends to touch a specific spot. Then I tried to alter the original code based on my request, then it did not work.

My wrong code:

```
#include <Adafruit_CircuitPlayground.h>
uint8_t pixeln = 0;

int capSensorPins[] = { 3, 2, 0, 1, 6, 9, 10 };

int numCapSensors = 7;

int ledForCap[] = { 0, 1, 3, 4, 6, 8, 9 };

int numNotes = 48;
int STmelody[] = {
  131, 165, 196, 247, 262, 247, 196, 165, 131, 165, 196, 247, 262, 247,
  196, 165, 131, 165, 196, 247, 262, 247, 196, 165, 131, 165, 196, 247, 262,
  247, 196, 165, 131, 165, 196, 247, 262, 247, 196, 165, 131, 165, 196, 247,
  262, 247, 196, 165
};
int STnoteDurations[] = {
```

```
  188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188,
188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188,
188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188,
188, 188, 188, 188
};

int THRESHOLD = 1000;

void setup() {
  Serial.begin(9600);
  CircuitPlayground.begin();
}

void loop() {
for (int i = 0; i < numCapSensors; i++)

{
Serial.print("Capsense #");

Serial.print(capSensorPins[i]);

Serial.print(": ");

Serial.println(CircuitPlayground.readCap(capSensorPins[i]));
  // play a melody if there is a touch, as an example
  // we'll pick pin 3, which is labeled SCL/A4 on the board
  // if the value is large enough
if (CircuitPlayground.readCap(3) > THRESHOLD)
    (CircuitPlayground.readCap(capSensorPins[i]) > THRESHOLD)
    (CircuitPlayground.setPixelColor(ledForCap[i],
CircuitPlayground.colorWheel(25 * i)));
    // play the melody (only if the slide switch is on)
    playMelody();
  delay(50);
 else

 CircuitPlayground.setPixelColor(ledForCap[i], 0, 0, 0);
}
// Play a note of the specified frequency and for the specified duration.
// No tones will play if the slide switch is in the -/off position.
```

```
void playNote(int frequency, int duration) {
  // Check if the slide switch is off
  if (!CircuitPlayground.slideSwitch()) {
    // stop immediately without playing anything.
    return;
  } else {
    // Play the note for the specified duration.
    CircuitPlayground.playTone(frequency, duration, false);
    // wait for the note to play
    delay(duration + duration / 16);
  }
}


//plays stranger things theme
//collaboration with TA Claudia
void playMelody() {
  for (int i = 0; i < numNotes; i++) {
    playNote(STmelody[i], STnoteDurations[i]);
  }
}
```

Final code:

```
#include <Adafruit_CircuitPlayground.h>
uint8_t pixeln = 0;

int capSensorPins[] = { 3, 2, 0, 1, 6, 9, 10 };

int numCapSensors = 7;

int ledForCap[] = { 0, 1, 3, 4, 6, 8, 9 };

int numNotes = 48;
int STmelody[] = {
  131, 165, 196, 247, 262, 247, 196, 165, 131, 165, 196, 247, 262, 247,
196, 165, 131, 165, 196, 247, 262, 247, 196, 165, 131, 165, 196, 247, 262,
247, 196, 165, 131, 165, 196, 247, 262, 247, 196, 165, 131, 165, 196, 247,
262, 247, 196, 165
};
int STnoteDurations[] = {
```

```
  188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188,
188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188,
188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188, 188,
188, 188, 188, 188
};

int THRESHOLD = 800;

void setup() {
  Serial.begin(9600);
  CircuitPlayground.begin();
}

void loop() {
  for (int i = 0; i < numCapSensors; i++)

  {
    Serial.print("Capsense #");

    Serial.print(capSensorPins[i]);

    Serial.print(": ");

    Serial.println(CircuitPlayground.readCap(capSensorPins[i]));
    // play a melody if there is a touch

    // if the value is large enough
    if (CircuitPlayground.readCap(capSensorPins[i]) > THRESHOLD) {
      CircuitPlayground.setPixelColor(ledForCap[i],
CircuitPlayground.colorWheel(25 * i));
      playMelody();
      }
  }
}


//plays stranger things theme
//collaboration with TA Claudia
void playMelody() {
  for (int i = 0; i < numNotes; i++) {
```

```
      playNote(STmelody[i], STnoteDurations[i]);
  }
}
void playNote(int frequency, int duration) {
  // Check if the slide switch is off
  if (!CircuitPlayground.slideSwitch()) {
    // stop immediately without playing anything.
    return;
  } else {
    // Play the note for the specified duration.
    CircuitPlayground.playTone(frequency, duration, false);
    // wait for the note to play
    delay(duration + duration / 16);
  }
}
```

Reflection:

Self-efficacy was likely a driving force at the beginning of my project. As me
conceptualized and started building the touch-sensor doorbell, my confidence in my skills and
problem-solving abilities pushed you forward. However, during the process, I had difficulties
dealing with my code, and these challenges might have shaken this confidence at the beginning.
Instead of viewing these obstacles as barriers that couldn't be crossed, I proactively reached out
to the professor and saw them as opportunities to learn and grow. Asking for help from your
professor indicates a willingness to learn and improve, rather than a fixed mindset where
challenges are seen as dead-ends.

During moments of frustration or when encountering coding problems, self-regulation
helped me to stay focused and persevere, rather than giving up. Finally, the concept of
belongingness, particularly when I sought help from your professor, was evident. This act not
only provided me with technical assistance but also reinforced my connection to a learning
community, fostering a sense of belonging that is vital for academic success.