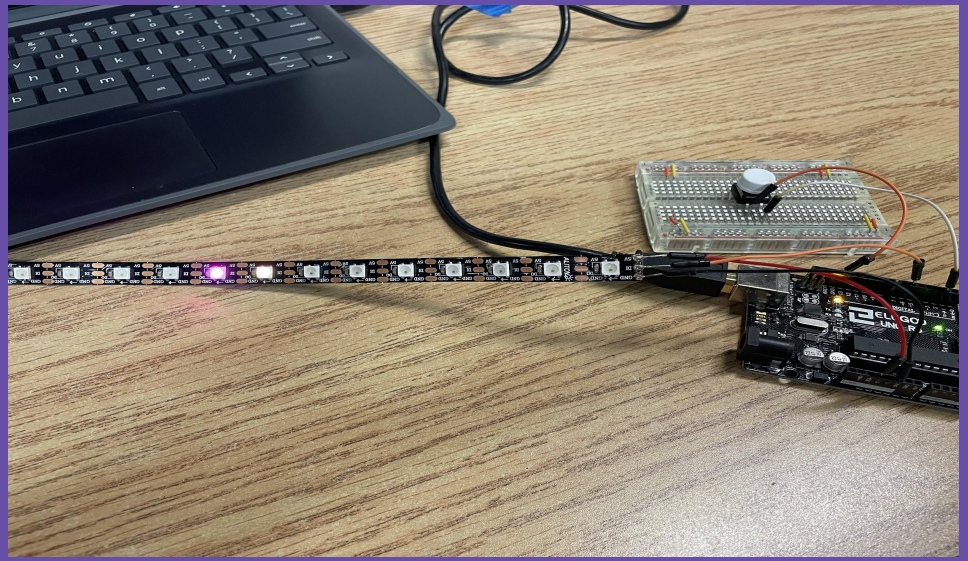
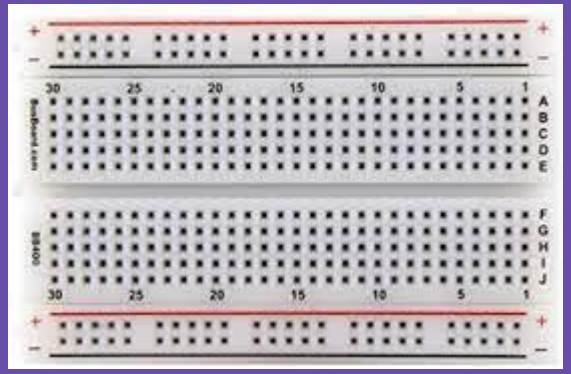
Computer Science Final

By Wiley Selvey



Materials Used

- Arduino Uno
- LED Strip
- Breadboard
- Button



The Game

The purpose of this code was to make a game where you would try to get the highest score you possibly can. To add to your score you must press the button when the player (pink light) is equal to the coins position (yellow light). One of the two coins is fake, and will change position each time the player passes it. If you press the button on the fake coin or a spot other than the real one than you lose one life. You have three lives, when you lose all three the game will reset itself. When you press the button on the coin it will change its position and flash white so you know which one it is. The final score will be showed in the monitor.

The link to the doc with the code is here
https://docs.google.com/document/d/1Xc226x0NB59EbbzdaMeMtCUsW4rOqQI4_ns1nscHh7M/edit

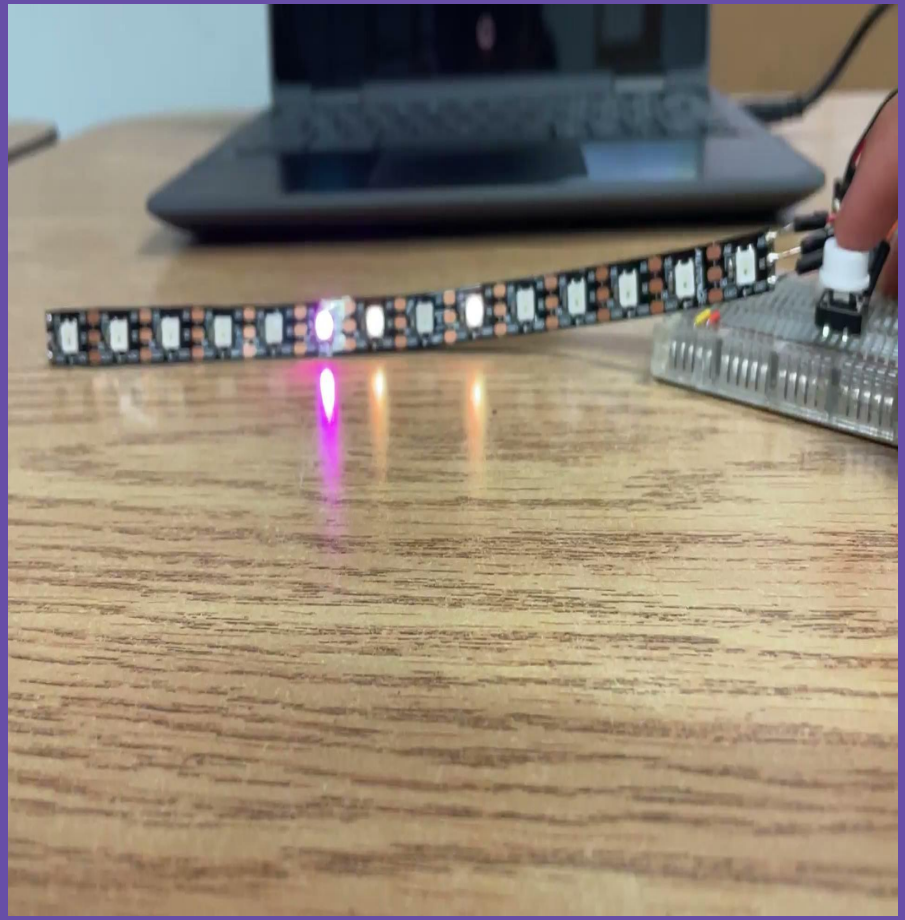
Example of the Code

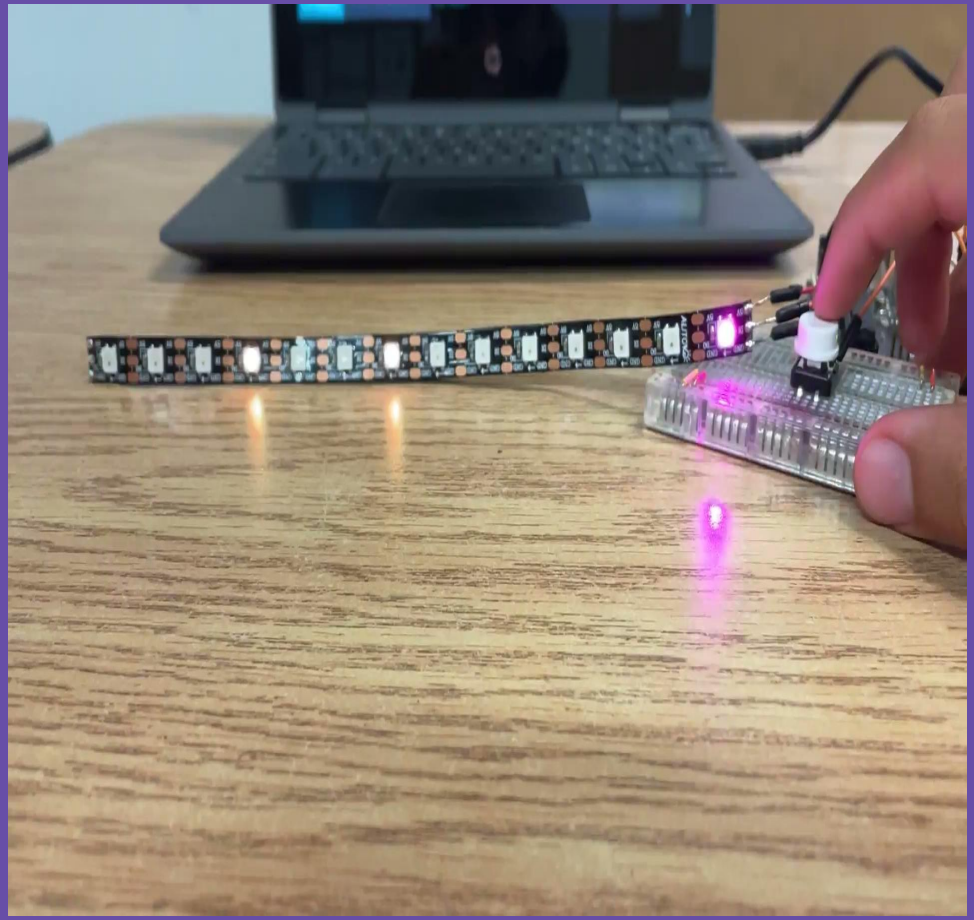
```
void checkClick() {
  if (clicked == true) {
    if (playerPos == coinPos) {
      score++;
      frameWait -= 0.2;
      coinPos = random(3, NUMPIXELS);
      coinFlash = true;
    } else {
      takeDamage = true;
    }
  }
}


void checkDamage() {
  if (takeDamage == true) {
    takeDamage = false;
    lives--; // each time you press the button when playerPos and coinPos are not equal you
    if (lives > 0) { // players loses life but doens't die
      strip.clear();
      strip.setPixelColor(playerPos - 1, 255, 0, 0);
      strip.show();
      playerPos = 0; // when player loses a life playerPos resets to neopixel 1
      delay(1000);
    } else { // player dies
      Serial.println(score); // prints the end score in the monitor
```

Checks if the button was clicked when the players position is equal to real coins position, if true than the score increases by one, the coins position changes, and flashes

If the button is clicked when the player and real coin is not equal than the player takes damage

When the player is out of lives its position resets to neopixel 0 and the final score is printed in the serial monitor





Wiring / Setup

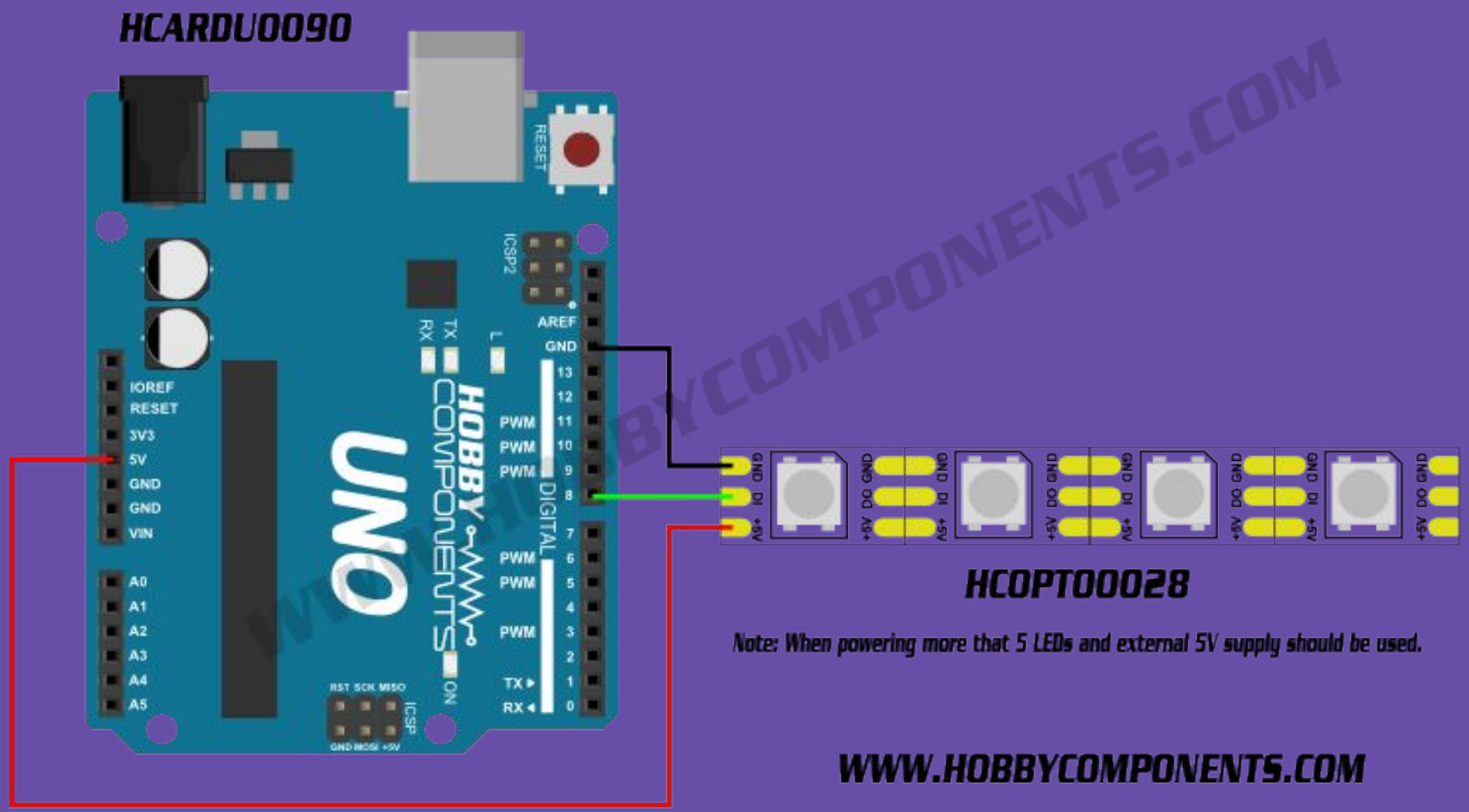
Step 1. Plug the USB cable into the Arduino and into theChromebook

Step 2. Get your LED strip, look under where the wires come out and plug each pin into its own designated socket on the Arduino :

GND → GND

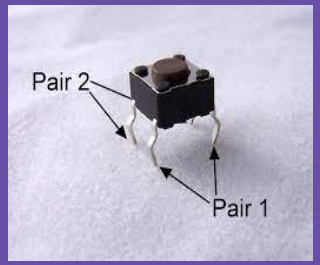
DI → whatever socket pin it says in your code, for example the images is pin 8

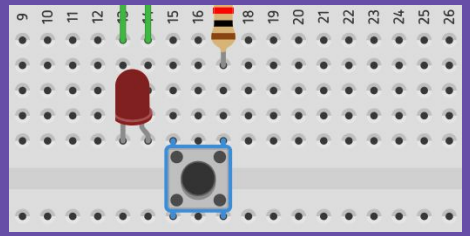
5V → either into 5V or 3.3V

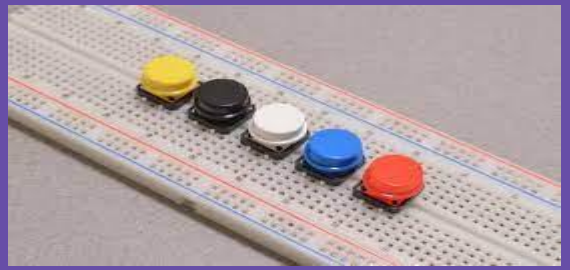

HCARDU0090
RESET
ICSP2
AREF
GND
13
12
PWM
11
PWM
10
PWM
9
8
DIGITAL
PWM
7
6
PWM
5
4
PWM
3
2
TX
1
RX
0
TX
RX
L
IOREF
RESET
3V3
5V
GND
GND
VIN
A0
A1
A2
A3
A4
A5
HOBBY COMPONENTS
UNO
ON
RST SCK MISO
ICSP
GND MOSI +5V
GND DI +5V
GND DO +5V
HCOPT00028
Note: When powering more that 5 LEDs and external 5V supply should be used.
WWW.HOBBYCOMPONENTS.COM

Wiring / Setup

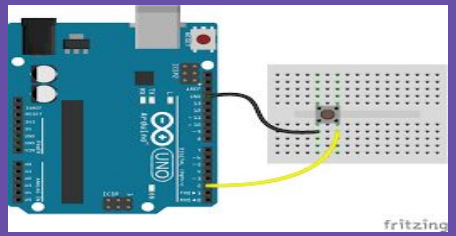
Step 3. Get your breadboard and your button. Plug your buttons pins into the breadboard with 2 pair of pins in each side.







Step 4. Use 2 wires to plug into the breadboard that are on the same row of one pair of buttons pins into GND and the other into whatever button pin your code says. For example, my button pin is 2.




```
#include <Adafruit_NeoPixel.h>
#define BUTTON_PIN   2 // A1 AD 2 Gnd
#define PIN    6
#define NUMPIXELS 14
```

The End