Benjamin Schmid: 169042790

Exploration Project

Overview:

The main goal of the project was to create a simple and intuitive distance measurement device using an ultrasonic distance sensor and an RGB led. The target user are individuals who require a visual prompt for positioning within a specific position such as a parking aid or art installations. Implementation of the RGB LED is as follows:

- **GREEN**: Motion towards the sweet spot
- **YELLOW:** Approaching the sweet spot, continue to motion towards and be ready to stop
- **RED:** User should stop as they are in the sweet spot
- **BLUE (Blinking):** user has moved past the sweet spot, user should move backward.

Packages and Libraries needed:

- **RPi.GPIO:** required for configuring the GPIO (General Purpose Input/Output) pins used to connect and control the ultrasonic distance sensor.
- **Time and Datetime:** required for proper functionality of ultrasonic distance sensor
  - Used to calculate travel time and pausing to give sensor time to send and receive echo signal
- **Gpiozero** (New)**:** required to treat the RGB LED as one LED rather than 3 different LED completely independent of eachother
  - streamlined the code and made it much more readable with far less lines than required
  - Specifically used RGBLED() function

- **Colorzero** (New)**:** used to streamline the color selection for said LED
    - Instead of inputting various color codes to the LED, the Color() function allows for common named colors to be inputted into the function
        - For example, Color('yellow') would change the LED to be yellow.
- **Threading:** Used to create a thread to wait for the stop command to be pressed.
    - Utilizing multithreading allows easy and proper exiting of the program

Challenges or Issues:

A major hurdle in the project came when wiring the LED. Generally, when wiring an LED, the resistor you chose does not really matter all that much if it is around 500 ohms. However, different color of LEDs have different forward voltages. Green and Blue tend to be around 3V where Red is around 2V. Using all the same resistors would result in different colors being brighter than the others. This is an issue when inputting color codes as the colors would not be accurate. For example, green being brighter than red would result in a greenish yellow when inputting color code FFFF00 (yellow)

Sources:

- https://www.instructables.com/Using-a-RPi-to-Control-an-RGB-LED/
    - Useful for understanding how to wire up the RGB led
- https://ledcalculator.net/
    - Useful for calculating the resistance required for the RGB LED pins.
- https://www.inolux-corp.com/datasheet/SMDLED/Addressable%20LED/IN-PI554FCH.pdf
    - Data sheet for the RGB LED, giving useful information such as forward voltage and current

- https://gpiozero.readthedocs.io/en/stable/api_output.html
    - Documentation for gpiozero library, outlining the different functions included with RGBLED
        - I used RGBLED() and blink()
- https://colorzero.readthedocs.io/en/release-1.0/api_color.html
    - Documentation for colorzero library, outlining the different functions
        - Color() is the one I used within my code.