```python
import RPi.GPIO as GPIO
import fileWriter

def turn_off(led):
    if GPIO.input(led):
        print("Turn off: " + str(led))
        GPIO.output(led, False)


def turn_on(led):
    if not GPIO.input(led):
        print("Turn on: " + str(led))
        GPIO.output(led, True)


class Button:

    def __init__(self, name, button_id, led1, led2, led3):
        self.name = name
        self.buttonId = button_id
        self.led1 = led1
        self.led2 = led2
        self.led3 = led3
        self.isRunning = False
        self.didPress = False
        self.holdLength = 0
        self.state = 0
        self.HOLD_THRESHOLD = 16

    def setup(self):
        if self.isRunning:
            return

        self.isRunning = True

        GPIO.setup(self.led1, GPIO.OUT)
        GPIO.setup(self.led2, GPIO.OUT)
        GPIO.setup(self.led3, GPIO.OUT)

        GPIO.setup(self.buttonId, GPIO.IN,
pull_up_down=GPIO.PUD_DOWN)

        turn_off(self.led1)
        turn_off(self.led2)
        turn_off(self.led3)

    def check_state(self):

        if GPIO.input(self.buttonId):
            if not self.didPress:
                self.didPress = True
                self.holdLength = 0
                print(self.name + " pressed, moving to state: " + str(self.state))
        elif self.didPress:
            if self.holdLength != -1:
                self.state += 1
            self.didPress = False
            self.holdLength = 0
            print(self.name + " released")

        #  Increment the holdLength if the button is held
        if self.didPress and self.holdLength != -1:
            self.holdLength += 1

        #  Check if we have been holding longer than the threshold
        if self.holdLength >= self.HOLD_THRESHOLD:
            # Write the amount of pills taken
            fileWriter.save_to_file(self.name, self.state)
            self.holdLength = -1
            self.state = 0
            print(self.name + " reset by holding")

        #  Turn on/off leds based on state
        if self.state >= 3:
            turn_on(self.led3)
            self.state = 3

        if self.state >= 2:
            turn_on(self.led2)

        if self.state >= 1:
            turn_on(self.led1)

        if self.state == 0:
            turn_off(self.led1)
            turn_off(self.led2)
            turn_off(self.led3)

    def get_name(self):
        return self.name

```

```
Program Started 16:29:55 2022-11-22

16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 1
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 0
16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 2
16:29:55 2022-11-22 :: Green 3
16:29:55 2022-11-22 :: Blue 1
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 1
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 2
16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 1
16:29:55 2022-11-22 :: Green 1
16:29:55 2022-11-22 :: Blue 3
16:29:55 2022-11-22 :: Red 1
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 0
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 0
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 2
Program Started 16:29:55 2022-11-22

Program Started 16:29:55 2022-11-22

16:29:55 2022-11-22 :: Red 3
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 3
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 2
16:29:55 2022-11-22 :: Green 3
16:29:55 2022-11-22 :: Blue 0
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 1
```

```
16:29:55 2022-11-22 :: Blue 0
Program Started 16:29:55 2022-11-22

16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 1
16:29:55 2022-11-22 :: Green 1
16:29:55 2022-11-22 :: Blue 2
16:29:55 2022-11-22 :: Red 3
16:29:55 2022-11-22 :: Yellow 0
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 3
16:29:55 2022-11-22 :: Red 3
16:29:55 2022-11-22 :: Yellow 0
16:29:55 2022-11-22 :: Green 3
16:29:55 2022-11-22 :: Blue 0
16:29:55 2022-11-22 :: Red 1
16:29:55 2022-11-22 :: Yellow 0
16:29:55 2022-11-22 :: Green 3
16:29:55 2022-11-22 :: Blue 1
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 1
16:29:55 2022-11-22 :: Green 3
16:29:55 2022-11-22 :: Blue 1
16:29:55 2022-11-22 :: Red 3
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 3
16:29:55 2022-11-22 :: Blue 2
Program Started 16:29:55 2022-11-22

16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 2
16:29:55 2022-11-22 :: Green 2
16:29:55 2022-11-22 :: Blue 0
16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 0
16:29:55 2022-11-22 :: Green 1
16:29:55 2022-11-22 :: Blue 2
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 3
16:29:55 2022-11-22 :: Blue 2
```

```
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 2
16:29:55 2022-11-22 :: Green 1
16:29:55 2022-11-22 :: Blue 2
16:29:55 2022-11-22 :: Red 1
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 3
16:29:55 2022-11-22 :: Blue 3
16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 1
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 1
16:29:55 2022-11-22 :: Green 2
16:29:55 2022-11-22 :: Blue 3
16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 2
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 2
16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 1
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 0
Program Started 16:29:55 2022-11-22

16:29:55 2022-11-22 :: Red 1
16:29:55 2022-11-22 :: Yellow 2
16:29:55 2022-11-22 :: Green 1
16:29:55 2022-11-22 :: Blue 2
16:29:55 2022-11-22 :: Red 3
16:29:55 2022-11-22 :: Yellow 1
16:29:55 2022-11-22 :: Green 3
16:29:55 2022-11-22 :: Blue 0
16:29:55 2022-11-22 :: Red 3
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 3
16:29:55 2022-11-22 :: Red 3
16:29:55 2022-11-22 :: Yellow 0
16:29:55 2022-11-22 :: Green 0
```

```
16:29:55 2022-11-22 :: Blue 2
16:29:55 2022-11-22 :: Red 1
16:29:55 2022-11-22 :: Yellow 0
16:29:55 2022-11-22 :: Green 1
16:29:55 2022-11-22 :: Blue 3
16:29:55 2022-11-22 :: Red 3
16:29:55 2022-11-22 :: Yellow 1
16:29:55 2022-11-22 :: Green 1
16:29:55 2022-11-22 :: Blue 1
Program Started 16:29:55 2022-11-22

16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 0
16:29:55 2022-11-22 :: Green 1
16:29:55 2022-11-22 :: Blue 0
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 1
16:29:55 2022-11-22 :: Blue 0
16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 1
16:29:55 2022-11-22 :: Blue 1
Program Started 16:29:55 2022-11-22

Program Started 16:29:55 2022-11-22

Program Started 16:29:55 2022-11-22

16:29:55 2022-11-22 :: Red 1
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 1
16:29:55 2022-11-22 :: Red 2
16:29:55 2022-11-22 :: Yellow 0
16:29:55 2022-11-22 :: Green 2
16:29:55 2022-11-22 :: Blue 3
16:29:55 2022-11-22 :: Red 1
16:29:55 2022-11-22 :: Yellow 1
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 2
```

```
16:29:55 2022-11-22 :: Red 1
16:29:55 2022-11-22 :: Yellow 2
16:29:55 2022-11-22 :: Green 2
16:29:55 2022-11-22 :: Blue 1
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 3
16:29:55 2022-11-22 :: Green 3
16:29:55 2022-11-22 :: Blue 2
16:29:55 2022-11-22 :: Red 0
16:29:55 2022-11-22 :: Yellow 2
16:29:55 2022-11-22 :: Green 0
16:29:55 2022-11-22 :: Blue 0
```

```python
from datetime import date
from datetime import datetime

output_file = 'output.log'


# saves a pill counters value to file
def save_to_file(light, val):
    file = open(output_file, "a")  # open file in append

    out = str(datetime.now().strftime("%H:%M:%S")) + ' ' + str(date.today()) + ' :: ' + str(light) + ' ' + str(val) + '\n'

    file.write(str(out))  # write out to file
    file.close()  # close file


# appends a star message to the output file
def announce_start():
    file = open(output_file, "a")  # open file in append

    out = 'Program Started ' + str(datetime.now().strftime("%H:%M:%S")) + ' ' + str(date.today()) + '\n\n'

    file.write(str(out))  # write out to file
    file.close()  # close file
```

```python
import RPi.GPIO as GPIO

import fileWriter
from Button import Button
import time

fileWriter.announce_start()

GPIO.setmode(GPIO.BOARD)

blueButton = Button("Blue", 13, 7, 10, 8)
greenButton = Button("Green", 19, 15, 18, 16)
yellowButton = Button("Yellow", 36, 21, 24, 23)
redButton = Button("Red", 31, 40, 29, 26)

blueButton.setup()
greenButton.setup()
yellowButton.setup()
redButton.setup()

while (True):
    blueButton.check_state()
    greenButton.check_state()
    yellowButton.check_state()
    redButton.check_state()
    time.sleep(.125)

```