Chris Barr
Github root directory: https://github.com/BarrChris/submissions_da_F19

Date Submitted: 09/24/19

Task 00: Execute provided code

Youtube Link: https://www.youtube.com/watch?v=Omlmqvm73Zw

Task 01:

Youtube Link: https://www.youtube.com/watch?v=jui3c_4R82Y

Modified Schematic (if applicable):
N/A

Modified Code:

```
#include <stdint.h>
#include <stdbool.h>
#include "inc/hw_types.h"
#include "inc/hw_memmap.h"
#include "driverlib/sysctl.h"
#include "driverlib/gpio.h"
int main(void)
{
        uint8_t ui8LED = 2;
        SysCtlClockSet(SYSCTL_SYSDIV_4|SYSCTL_USE_PLL|SYSCTL_XTAL_16MHZ|SYSCTL_OSC_M
AIN);
        SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);
        GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3);
        while(1)
        {
            // Turn on the LED
            GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, ui8LED);
            // Delay for a bit
            SysCtlDelay(8000000);
            // Cycle through Red, Green and Blue LEDs
            if (ui8LED == 8) {ui8LED = 2;} else {ui8LED = ui8LED*2;}
        }
}
```

Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.

Chris Barr
Github root directory: https://github.com/BarrChris/submissions_da_F19

Task 02a:

Youtube Link: https://www.youtube.com/watch?v=TE_021JIkcI

Modified Schematic (if applicable):
N/A

Modified Code:

```
#include <stdint.h>
#include <stdbool.h>
#include "inc/hw_memmap.h"
#include "inc/hw_types.h"
#include "driverlib/sysctl.h"
#include "driverlib/gpio.h"

uint8_t ui8PinData=4;
uint8_t RGB_BGR=0;
uint8_t count=0;

int main(void)
{
    SysCtlClockSet(SYSCTL_SYSDIV_5|SYSCTL_USE_PLL|SYSCTL_XTAL_16MHZ|SYSCTL_OSC_MAIN);

    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);
    GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3);

    while(1)
    {
            //Change color, and turn off to get ready to blink to next color
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, ui8PinData);
        SysCtlDelay(8000000);
        GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0x00);
        SysCtlDelay(8000000);

            //If still going through BGR sequence, go here
        if (count < 2)
        {
            //red = 2, blue = 4, green = 8
            //this will do blue, green, red; 4, 8, 2
            if(ui8PinData==4)
                ui8PinData=8; //green
            else if (ui8PinData==8)
                ui8PinData=2; //red
            count++;
        }

            //Else, if finished with BGR sequence, go here for the RBG sequence
        else if (count > 1)
        {
            //this will do red, green, blue; 2, 8, 4
            if(count == 2)
```

```
                ui8PinData=2; //red
            else if (count == 3)
                ui8PinData=8; //green
            else
                ui8PinData=4; //blue

            count++;

                    //Reset everything once the sequence is over
            if (count > 5)
            {
                count = 0;
            }
        }
    }
}
```

Chris Barr
Github root directory: https://github.com/BarrChris/submissions_da_F19

Task 02b:

Youtube Link: https://www.youtube.com/watch?v=whZbuqhJbxw

Modified Schematic (if applicable):
N/A

Modified Code:

```
#include <stdint.h>
#include <stdbool.h>
#include "inc/hw_memmap.h"
#include "inc/hw_types.h"
#include "driverlib/sysctl.h"
#include "driverlib/gpio.h"

uint8_t ui8PinData=4;
uint8_t RGB_BGR=0;
uint8_t count=0;

//function prototype(s)
void blink(uint8_t);
void delay(void);

int main(void)
{
    SysCtlClockSet(SYSCTL_SYSDIV_5|SYSCTL_USE_PLL|SYSCTL_XTAL_16MHZ|SYSCTL_OSC_MAIN);

    SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOF);
    GPIOPinTypeGPIOOutput(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3);

    //red = 2, blue = 4, green = 8
    while(1)
    {
        ui8PinData=2; //Red
        blink(ui8PinData);
        delay();

        ui8PinData=8; //Green
        blink(ui8PinData);
        delay();

        ui8PinData=4; //Blue
        blink(ui8PinData);
        delay();

        ui8PinData=10; //Yellow
        blink(ui8PinData);
        delay();
```

Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.

```
        ui8PinData=6; //Purple
        blink(ui8PinData);
        delay();

        ui8PinData=12; //Cyan
        blink(ui8PinData);
        delay();

        ui8PinData=14; //White
        blink(ui8PinData);
        delay();
    }
}

//Blinks to specific color with delay
void blink(uint8_t ui8PinData)
{
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, ui8PinData);
    SysCtlDelay(8000000);
}

//Turns off LED with delay
void delay(void)
{
    GPIOPinWrite(GPIO_PORTF_BASE, GPIO_PIN_1|GPIO_PIN_2|GPIO_PIN_3, 0x00);
    SysCtlDelay(8000000);
}
--------------------------------------------------------------------------------
```