```
/**
  Generated Main Source File

  Company:
    Microchip Technology Inc.

  File Name:
    main.c

  Summary:
    This is the main file generated using PIC10 / PIC12 / PIC16 / PIC18 MCUs

  Description:
    This header file provides implementations for driver APIs for all modules selected in the
GUI.
    Generation Information :
        Product Revision:  PIC10 / PIC12 / PIC16 / PIC18 MCUs - 1.81.7
        Device:  PIC18F26K20
        Driver Version:  2.00
*/

/*
    (c) 2018 Microchip Technology Inc. and its subsidiaries.

    Subject to your compliance with these terms, you may use Microchip software and any
    derivatives exclusively with Microchip products. It is your responsibility to comply with third
party license terms applicable to your use of third party software (including open source
software) that may accompany Microchip software.

THIS SOFTWARE IS SUPPLIED BY MICROCHIP "AS IS". NO WARRANTIES, WHETHER
EXPRESS, IMPLIED OR STATUTORY, APPLY TO THIS SOFTWARE, INCLUDING ANY
IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS
FOR A PARTICULAR PURPOSE.

IN NO EVENT WILL MICROCHIP BE LIABLE FOR ANY INDIRECT, SPECIAL, PUNITIVE,
INCIDENTAL OR CONSEQUENTIAL LOSS, DAMAGE, COST OR EXPENSE OF ANY
KIND WHATSOEVER RELATED TO THE SOFTWARE, HOWEVER CAUSED, EVEN IF
MICROCHIP HAS BEEN ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE
FORESEEABLE. TO THE FULLEST EXTENT ALLOWED BY LAW, MICROCHIP'S TOTAL
LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO THIS SOFTWARE WILL NOT
EXCEED THE AMOUNT OF FEES, IF ANY, THAT YOU HAVE PAID DIRECTLY TO
MICROCHIP FOR THIS SOFTWARE.
*/

#include "mcc_generated_files/mcc.h"

/*
                         Main application
```

```
 */

char data;

void send_string(const char *x){
   while (*x){
   EUSART_Write(*x++);
   }
}

void main(void)
{
   // Initialize the device
   SYSTEM_Initialize();

   // If using interrupts in PIC18 High/Low Priority Mode you need to enable the Global High
and Low Interrupts
   // If using interrupts in PIC Mid-Range Compatibility Mode you need to enable the Global
and Peripheral Interrupts
   // Use the following macros to:

   // Enable the Global Interrupts
   //INTERRUPT_GlobalInterruptEnable();

   // Disable the Global Interrupts
   //INTERRUPT_GlobalInterruptDisable();

   // Enable the Peripheral Interrupts
   //INTERRUPT_PeripheralInterruptEnable();

   // Disable the Peripheral Interrupts
   //INTERRUPT_PeripheralInterruptDisable();

    char * welcome_msg1= "Welcome to CO326 Lab3";
   char * welcome_msg2= "Press 1 for Red, 2 for Yellow, 3 for Green ... \n";
   char * add_line="\n \r";

   // welcome messages on startup
   send_string(welcome_msg1);
   send_string(add_line);
   __delay_ms(500); // delay of 500ms
   send_string(welcome_msg2);
   send_string(add_line);

   while (1)
   {
      //read input
      data=EUSART_Read();
```

```
        //input checking
        switch (data) {
            case '1':
                Red_SetHigh();
                break;

            case '2':
                Yellow_SetHigh();
                break;

            case '3':
                Green_SetHigh();
                break;

            default:
                Red_SetLow();
                Yellow_SetLow();
                Green_SetLow();
                break;
        }
    }
}
/**
 End of File
*/
```