Analyze the code sample provided in the appendix below to answer the following questions.

1. Summarize, at a high level, what the program does.

At a high level the program lights up the LED then the program asks the user for a number. Then program takes that number and subtracts it from 255 and converts it to HEX. This is the amount of brightness it displays.

1. Explain, in more detail, what each of the specific code sections does. There are six code sections identified by comment lines and highlighted in blue.

Section 1: The LEDs are assigned a constant variable that cannot be changed through the program. redPin is assigned 3. greenPin is assigned 5. bluePin is assigned 6. It also sets up the program.

Section 2: Changes the speed so that this program would work. Serial.begin(9600); . Sets all the pins to output and not input. Creates a loop for the program for whatever is in the { } brackets underneath it. IT also waits until the user puts something in the monitor part to respond to the program.

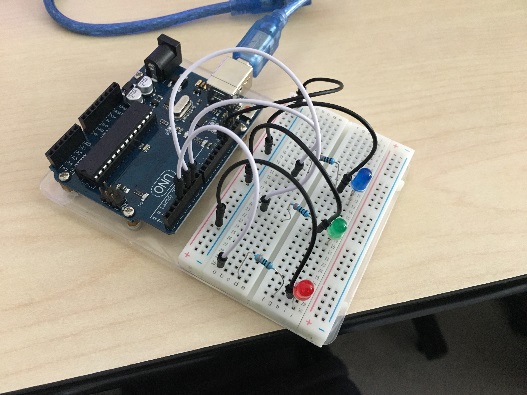
Section 3: The colors are assigned a number from the user from section to. The same value is given to all the pins. The program reads the number that the user entered and creates a new line.

Section 4: Makes the calculations by setting the LED to 255 then subtracts the number which was inputted by the user then the LED is ready to be set to a certain brightness.

Section 5: The LED is actually set to the brightness.

Section 6: It displays on the serial monitor the color and the hexadecimal value.

1. Draw a diagram of the wiring diagram for the connection of LEDs to the Arduino board. Make sure to label and identify all pin numbers and assignments.



1. List all of the outputs of the program. Use a table similar to what you did in Module B.1. Make a table listing all of the outputs and their associated meaning.

|  |  |
| --- | --- |
| Output | Meaning |
| Pin 3 | Turn LED on/off/changes brightness |
| Pin 5 | Turn LED on/off/changes brightness |
| Pin 6 | Turn LED on/off/changes brightness |
| Blue, red, green and hex value | Changes brightness of LED from decimal to HEX and name of each color. |

1. List all of the inputs to the program. Use a table similar to what you did in Module B.1. Make a table listing all of the inputs and their associated action.

|  |  |
| --- | --- |
| Input | Meaning |
| Inputing a number in to the serial monitor | Dims the LED from 255 by inputted value capped at 255 and minimum at 0. |

1. Provide an example of console input that would cause the program not to work properly. (i.e. Input that would cause an error.)  
     
   A string value that is not a integer for example “red” “blue”.

Start of Code Appendix

**// Code Section 1:**

const int redPin = 3;

const int greenPin = 5;

const int bluePin = 6;

void setup() {

**// Code Section 2:**

Serial.begin(9600);

pinMode(redPin, OUTPUT);

pinMode(greenPin, OUTPUT);

pinMode(bluePin, OUTPUT);

}

void loop() {

while (Serial.available() > 0) {

**// Code Section 3:**

int red = Serial.parseInt();

int green = Serial.parseInt();

int blue = Serial.parseInt();

// The character '\n' is a newline character appended to the typed in message   
 // from the serial console.

if (Serial.read() == '\n') {

**// Code Section 4:**

red = 255 - constrain(red, 0, 255);

green = 255 - constrain(green, 0, 255);

blue = 255 - constrain(blue, 0, 255);

**// Code Section 5:**

analogWrite(redPin, red);

analogWrite(greenPin, green);

analogWrite(bluePin, blue);

**// Code Section 6:**

Serial.print(red, HEX);

Serial.print(green, HEX);

Serial.println(blue, HEX);

}

}

}

\*\*\* End of Code Appendix