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

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

The programs takes input in the serial monitor and changes the brightness of the 3 LEDs that are connected to 3, 5, 6. Then the program would print the number value for the brightness in hexadecimal in the serial monitor.

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: It sets the integer values for the variables and the values cannot be changes because it is constant variables. These variables are for the pins. This also starts the program.

Section 2: This section sets the amount of bytes for the serial monitor. This also sets the pins as outputs for redPin, greenPin and bluePin. This means that the LEDs are set to light up as outputs. Then the loop starts and waits for an input.

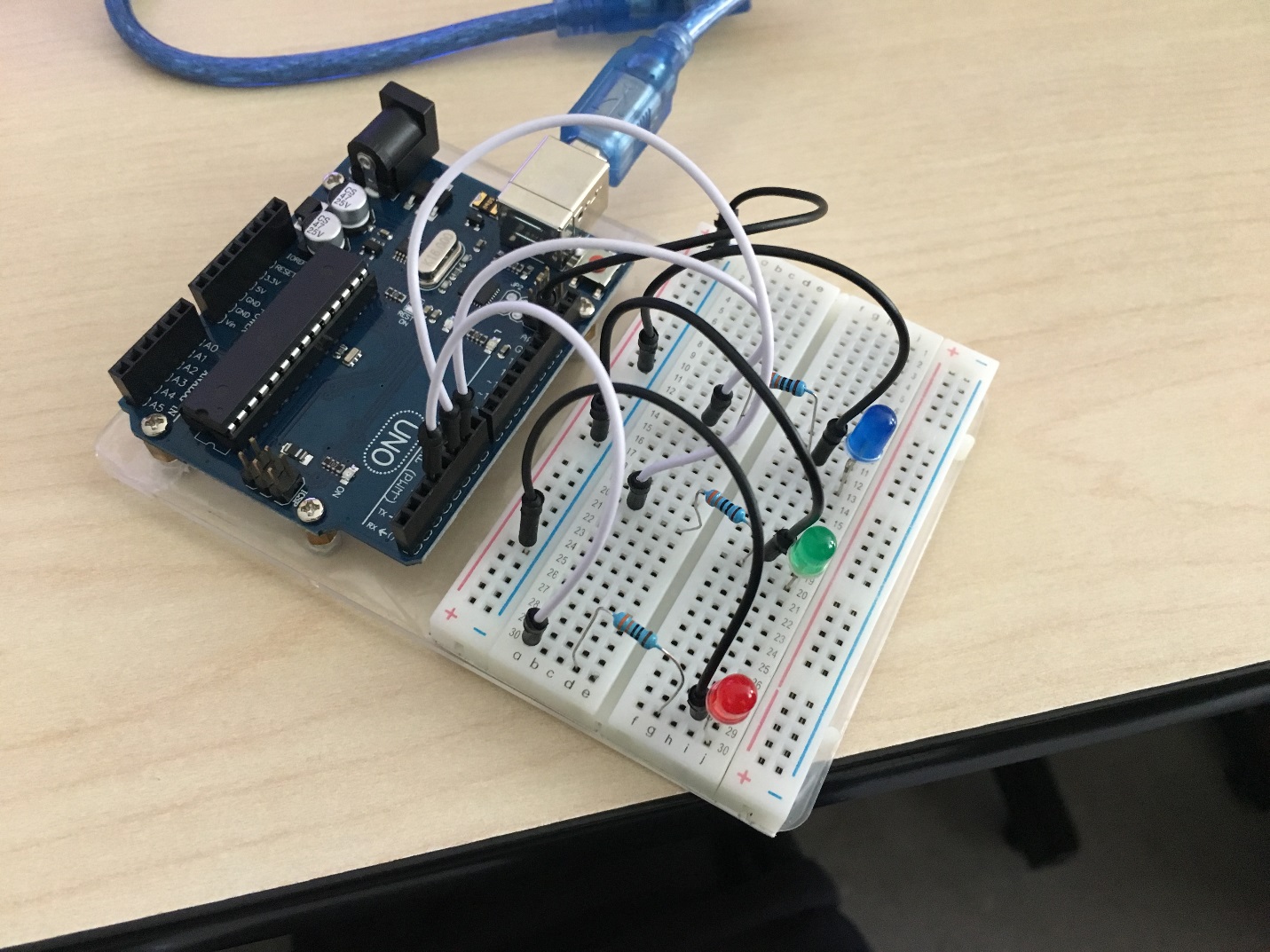
Section 3: This section asks for an integer input for red, green and blue. Then it would check for a new line and then continues the program.

Section 4: For the variables red, green and blue it would take the numbers that you inputted in section 3 and puts it in a number range for the program to choose a number. Then it would subtract the number chosen from 255.

Section 5: This section would make the LEDs light to the brightness value that was chosen.

Section 6: This sections prints the hexadecimal value of the brightness of the LEDs in the serial monitor.

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 | Turns the LED on/off/changes the brightness. |
| Pin 5 | Turns the LED on/off/changes the brightness. |
| Pin 6 | Turns the LED on/off/changes the brightness. |
| Red, Green, Blue and Hex values | Changes the brightness of the LEDs and converts the brightness value to hexadecimal. |

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** |
| Inputting numbers into serial monitor | The program reads the integer value you inputted and changes the brightness of the LED. |

1. Provide an example of console input that would cause the program not to work properly. (i.e. Input that would cause an error.)

If you input any string that is not a number, the program would not work. For example, if you write the string “red” it would cause the program not to work.

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