LEVEL 0

{digitalWrite(i, LOW); // Turn OFF the I/O board LEDdelay(200); }for (int i = BASE; i < BASE + NUM; i ++){digitalWrite(i, HIGH); // Turn ON the I/O board LEDdelay(200); }

**Level 0: LED Trailing Effects**

1. Implement the lesson titled “LED Trailing Effects”.
2. Locate on-line documentation that describes the C language “for” loop.

the loop() function does precisely what its name suggests, and loops

consecutively, allowing your program to change and respond. Use it to actively

control the Arduino board.

Example:

int buttonPin = 3;

1. What is the index and how is it used?

A "For" Loop is used to repeat a part of the code for as many times as you want

For example, if you want to check the score of a robot in the competition you would loop from 1 to that number.

* 1. When does the for loop end?

The for loop runs only one time which you can also call a cycle, when the program ends the for loop ends.

* 1. How is a “for” loop different from a “while” and a “do” loop?

A do/while loop will always execute the code in the do{} block first and then evaluate the condition. While for loop allows you to initiate a counter variable, a check condition, and a way to increment your counter all in one line

1. Research the “<” Comparitor.
   1. List all the other comparitors defined for the C language.
      * Standard C library provides qsort() that can be used for sorting an array the function uses QuickSort algorithm to sort the given array.
   2. Modify the “for” loop to use the “<=” comparator

<= (less than or equal to) is used in conjunction with a comparison operator, tests whether a certain condition has been reached, such as an input being above a certain number.

1. Research the “++” incrementor operator.
   1. Explain how this is different from the “=+ 1” assignment

++ is a unary operator that can be applied to variables and increments the value they hold. For

example often for loops have, as their increment-expr something like counter++

* 1. Modify the “for” loop to use the “=+” assignment

+= (compound addition) Performs a mathematical operation on a variable with another constant or variable. The += (et al) operators are just a convenient shorthand for the expanded syntax, listed below.

**Modified code:**

int BASE = 2;

int NUM = 6;

void setup()

{

for (int i = BASE; i <= BASE + NUM; i ++)

{

pinMode(i, OUTPUT);      //set port ‘i’ as an output port

}

}

void loop()

{

for (int i = BASE; i <= BASE + NUM; i ++)

{

digitalWrite(i, LOW);      // Turn OFF the I/O board LED

delay(200);

}

for (int i = BASE; i <= BASE + NUM; i ++)

{

digitalWrite(i, HIGH);    // Turn ON the I/O board LED

delay(200);

}

}