EENG 498: Preliminary Demo: Arduino Rubric   Student 1 _____   Student 2 _____

	Excellent (100%)	Competent (60%)	Developing (30%)	Grade1	Grade 2
Documentation 5 points each element	• Document listing answers to all questions in the handout • Well commented and easily readable sketches for all challenges	• Missing answers to questions on handout. • Sketches with minimal comments and not well formatted/readable.	• Answers to most questions missing. • Sketches have no comments	___/15	___/15
Demonstration 5 points each element	• Well defined use of arrays in sketch for challenge 1. • Can explain how the code works, how arrays were utilized and what order LEDs will blink on and off. Variable names make sense. • Well-formatted and readable display of temperature on monitor using serial.print (challenge 3). • Can explain how the code works, answer questions pertaining to their sketch. Variable names make sense.	• Inefficient use of arrays in sketch for challenge 1. • Unsure of certain aspects of sketch maybe because code was borrowed from the web. • Unformatted (unreadable) display of temperature on monitor (challenge 3) • Can explain how parts of the sketch work. Borrowed code from the web but lack of understanding.	• Did not use arrays in sketch for challenge 1. • Sketch copied from web and cannot explain what it does or how it works. • Did not use serial.print for debugging purposes in challenge 3. • Cannot explain how challenge 3 was implemented. Variable names don't make sense.	___/20	___/20
Reflection 30 points	• Student 1: Reflection Log Completed. • Student 2: Reflection Log Completed	• Student 1: Reflection Log partially completed. • Student 2: Reflection Log partially completed	• Student 1: Reflection Log mostly incomplete. • Student 2: Reflection Log mostly incomplete.	___/30	___/30

Total Score Student 1 _____: _____/50   **Total Score Student 2 _____: _____/50**

Arduino Challenges

1. Hook up 4 LEDs to pins 2 – 5. Use arrays to turn on each one in order until all of them have lit up and extinguish them in order until every one of them is off.

3. Make a digital thermometer. Break the challenge down in to smaller tasks before putting it all together. Ensuring that each individual piece works makes debugging much easier.
   a. Use serial.print functions to keep track of what is happening in your code and debugging.
   b. Use the TMP36 chip as the temperature sensor and send the values to the serial monitor. More information can be found here.
   c. Interface an LCD screen with the Arduino. You will need to include the LCD library that provides functions to work with the LCD. You can make your own libraries as well if you need to, later on based on the kind of hardware you use. The LCD library is a standard library that comes pre-installed with the Arduino IDE. Hook up the LCD and follow the "Hello World" tutorial. Play around with other LCD tutorials if you need to.
   d. Bring everything together such that the temperature is displayed on the LCD in Celsius and Fahrenheit.