Introduction to Computer Science
Pumpkin Lab Report

Due:

In this lab you will be using hardware and software to create a spooky (or fun!) Halloween pumpkin decoration. This is a chance to be creative! You're encouraged to design your pumpkin however you like keeping in mind that all design must be school-appropriate and must not violate our school's diversity and inclusion guidelines. In some cases, students groups have even chosen to "think outside the pumpkin" if you have an idea for a non-pumpkin decoration that fits with the other requirements for the lab, please let me know – *all non-pumpkin projects require prior approval.*

This lab will be done primarily in class, in groups of two or three. You may choose your group. We will do many parts of this lab as a class and I'll come around to help groups that are having trouble. All steps with double borders require a check-off before you move on.

LAB STEP	**Sign-Off**
STEP 1: Set up your programming environment. You will need access to the Arduino IDE. You have the option either to use a cloud-based IDE (you will create an account to store your code), or to install the IDE on your home computer. Both options have benefits! The cloud based IDE makes it easy to work on your project from multiple computers. Installing on your machine makes it easy to work when you don't have access to an Internet connection. This page links to the cloud-based IDE and to instructions for installing on your personal machine. https://www.arduino.cc/en/Guide/HomePage If you need help decided which is the right option for you, please ask. We won't use the IDE until the second class, so you have some time to set up your work environment.	
STEP 2: Check out a kit of parts. This part contains all of the basic elements you will need to create your pumpkin masterpiece! We will go over each part and its role in the project in class. We will not use all the parts in your kit, but please do a quick inventory and make sure you have all the parts listed below. If not, see me for a replacement. Pictures of all parts are provided at the end of this report for reference. ☐ Arduino Uno board ☐ USB cable ☐ Jumper wires ☐ LEDs ☐ Resistors ☐ Breadboard ☐ 9V battery ☐ Battery Clip ☐ Ultrasonic sensor	

STEP 3: Build a simple circuit that powers the LED . We will do this part together in class. You should end with a circuit that includes your Arduino, the breadboard, a resistor, and one LED. Follow along with the class as we complete the steps together making sure both partners get a chance to work with the hardware. If you need to review the steps, the slide handout is posted in Google Classroom.	
STEP 4: Add a Second LED. In class we worked together to make a circuit that lit up one LED. Can you add a second LED and make both LEDs light up at the same time? Work with your partner to accomplish this challenge and call me over to check off your project when you have it finished.	
STEP 5: Make the LED Blink! Now we will write an Arduino program to make our LED blink. Make sure you follow along the in-class example as you will need to make a small change to your circuit before you write your code. If you need to review the steps, the slide handout is posted in Google Classroom. We will be learning the basics of writing programs in the Arduino language, including how to load our finished programs onto our Arduinos.	
STEP 6 Work with your partner to add a second blinking LED to your project. Can you make it blink opposite your first LED? (So when the first LED is on, the new one is off. When the first LED is off, the new one is on). Think about what you need to change in the wiring and in the code. Call me over to check off your project when you have it finished If you finish early try to make the LEDs blink in different patterns. Try adding additional LEDs too!	
STEP 7: Wire the sensor and write a program to read distance data from it. Follow along in class as we do this together.	
STEP 8: Combine your code from the LED parts of the lab with the sensor part from step 7. Work with your partner to figure out how to use the LED to indicate that something is “near” the pumpkin.	
STEP 9: Now you’re ready to design your pumpkin! You will need to create a pumpkin that reacts to a trick-or-treater approaching. The minimum requirements for your pumpkin are: • Use a distance sensor to detect when something is “near’ the pumpkin • Have a minimum of 2 LEDs that are used to indicate that something has been detected “near” the pumpkin • Sketch a design of your pumpkin using the page provided in your lab report. Be creative! Have fuN! • Carve design into you pumpkin using scissors (please wear safety glasses for this part) making sure to leave room for the sensor and LEDs. You may use any materials in lab to augment your design • Complete the lap report reflection questions	

OPTIONAL STEPS!

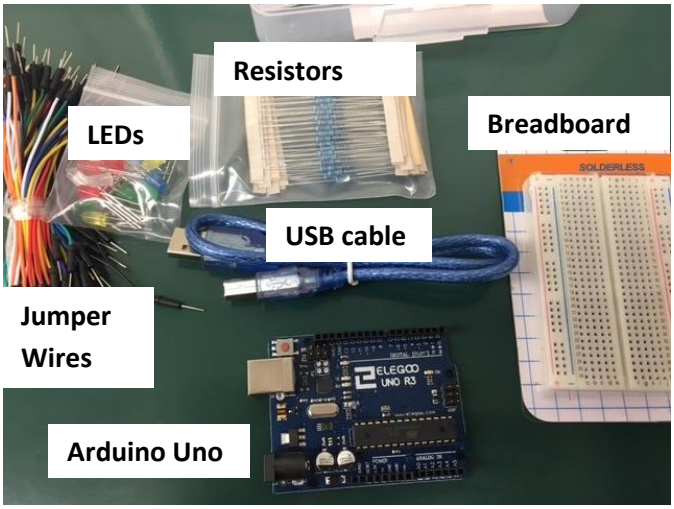
If you get your pumpkin working with the sensor and LEDs, you can add on to your project using other components we have in lab. In particular we have:
Speakers you can add a speaker to your circuit to play spooky sounds or music as part of your pumpkin design.
Micro Servos these are small motors that can be used to add movement to your projects

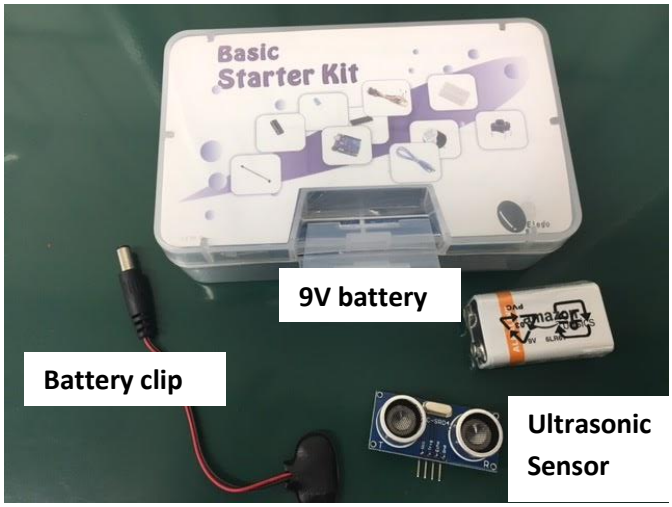
To add a speaker or micro servo(s) to your project, you will follow the same procedure you did for adding the distance sensor

(1) Wire the new component into your circuit
(2) Find a library that makes using the component easier
(3) Write code that tests the component individually to make sure you are sure how it works
(4) Integrate the new component into your project

If you choose to use any of the optional components, let me know and we can talk about next steps

Parts Reference





Use this page to sketch your pumpkin design. Make sure you think about where you will place the sensor and the LEDs in your design.

Your lab reflection should have the following sections

1. **Abstract**. This should be a short (2-3 sentence description of how your pumpkin works). Include a picture of the finished pumpkin.

2. **Materials**. Provide a complete list of the materials used to create your pumpkin

3. **Wiring**. Provide a picture or diagram of your wiring.

4. **Process Reflection**. Answer the following questions about the process of creating your pumpkin
    A. Describe one similarity and one difference between Snap! And Arduino programming.
    B. What was the hardest part of the Arduino project. Give one example of something that you had to debug or where the first thing you tried didn't work. Explain how you eventually got it to work
    C. Describe how the sensor values you read are used to create the output for your pumpkin. Are there ranges for which your pumpkin works best? Or ranges for which it does not work at all?
    D. Why did we use the NewPing library? Explain why in general we use libraries in programming. How did the NewPing library provide a layer of abstraction for our project?
    E. **Appendix: Code**. Copy and paste your code into an appendix at the end of your report - make sure that it is well commented, styled nicely (readable) and it is clear which parts of your code accomplish the different functions of your pumpkin

You lab report should be thorough, thoughtful, and nicely formatted. Make sure that you have proofread your report before you turn it in. Please turn in your lab report using Google Classroom