ENGI1020 – Lab 6 Logbook

|  |  |  |
| --- | --- | --- |
| **Name** | **Lab & Date** | **Input & Output Devices** |
| Anton Guaman | Lab 6  November 8, 2020 | Input: temperature sensor  Kill switch: Button  Output: Console and graph from imported module |

* 0 Preparation
  + - Summary of lab task in your own words
    - Summary of preparation (no need to repeat entirely)

In the lab preparation I selected the temperature test project. This lab preparation had three parts of the design. The first part was to modify the algorithm used in Lab 4 for a while loop. The while loop will run until the kill switch the button is pressed. Meanwhile inside the while loop tempertature values taken in an interval are appended in a list. For time we will do a similar thing. For the second part of the design we wrote a contract for function that will produce an average and return the # of samples taken. The last part of the design was to practice our graphing skills from a given data. It was a Distance vs time graph. The lab preparation also consisted of creating test cases for the design part 1 and part 2. For design part 1 the longer the waiting time the higher number of values is expected. For design part 2 to manipulate the input I will use an icepack to decrease the temperature and my finger to increase the temperature.

The lab task is to use a function that will have three parameters analogPin, digitalPin, and interval. This will allow us to receive input from the temperature sensor which will enter every reading in a list. Then we will use a kill switch to stop the reading which is the button. After this the imported module ﻿matplotlib.pyplot will allow us to graph a Temperature vs Time graph displaying the changes of temperature manipulated. Also using a for loop we need to calculate the average value of the input taken and print it in the console.

* 1 Implementation
  + 1.1 Converting Design to Implementation
    - Notes about how you took your design and implemented it
    - DO NOT just insert your code (though if it is useful to include lines of code to explain, you can)

There are preset global variables that will be the parameters of our functions.

﻿interval = 0.1 #time between samples in seconds

analogPin = 0 #Grove port for analog sensor

buttonPin = 5 #Grove port for button

dataCollection = [] #list to store data from analog

time = [] #stores the time duration

With these preset variables we can create a function called gatherinfo. Inside the function we create a local variable that will store the input value from the button. Also, we create a local variable called currentime with a value of 0.

﻿def gatherinfo(analogPin,buttonPin,interval):

buttonval= digital\_read(buttonPin)

currentTime = 0

Inside the function we will use a will loop that will stop when the button is pressed (that value will be 1). After that inside the loop we will append the currenTime which its value increases 0.1 seconds each time into the global variable time. We will also append the input gathered from the temperature sensor to the global variable dataCollection. This will execute every 0.1 seconds using the sleep(interval).

times.append(currentTime)

currentTime += 0.1

tempinput = temp\_celsius(0)

dataCollection.append(tempinput)

buttonval = digital\_read(buttonPin)---verifies that the button is ==0

sleep(interval)

For the second part of the code that requires the average from the inputs of the temperature sensor. We begin by setting a preset variable named sumvalues = to 0. This will store the sum of the values of all the temperatures. Then we use a for loop that is open for any amount of inputs. Finally, to calculate the average we divide the sumvalues/len(dataCollection) and store it in the variable Average.

sumvalues = 0

for i in dataCollection:

sumvalues += i

Average= sumvalues/ len(dataCollection)

We use the return command for the variables that store the information required.

return dataCollection, Average

For the last step of the code from the modules imported in the dot notation style will allow us to print the information we have collected.

plt.plot(times,dataCollection, label = "Average data" )--- This will print the time data in the x axis and the dataCollection in the y axis

plt.xlabel("Time (ms)")---this command will name the x axis

plt.ylabel("Temperature(C)") ---this command will name the y axis

* + 1.2. Errors Encountered While Implementing
    - Notes about errors encountered in Python interpreter or syntax problems
    - Include notes about solution
    - If useful, include names of errors

No errors while implementing.

* + 1.3 Details and Facts from Implementation to Remember
    - Details about concepts that you learned from implementation
    - Technical details about interpreter, Arduino equipment, engi1020 module related to implementation
    - Anything else you think will demonstrate your learning and will be useful to remember for future

I learned something new from the implementation which is from the modules we imported. I learned how to graph the information collected and how to name the axis of the graph.

* 2 Testing
  + 2.1. Test Plan and Results
    - Should be in table format!!

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test** | **Input Manipulation** | **Output**  **Expected** | **Output**  **Observed** | **Investigation** |
| **Test 1** | **How did you**  **change the**  **input?**  I did not change the input it will remain room temperature. | **What did you**  **expect to**  **observe?**  I expect to observe a linear line. | **What did you**  **observe?**  I observed a linear line. | **Did they match?**  **If not, what did**  **you do (can**  **reference**  **Section 2.2)**  Yes, they did match! |
| **Test 2** | I changed the input by putting my finger on top of the sensor. This will increase the temperature read by the sensor. | I expect to observe the line increase. | I observed the line increase in a zig zag line. | They sort of matched, but I figured out there is a problem with how the hardware reacts with my finger. Overall they did match. |
| **Test 3** | I changed the input by putting an ice pack close to the sensor. This will decrease the temperature read by the sensor. | I expect to observe the line decrease. | I observed the line decrease. | Yes, they did match! |

**Figure 1: Test 1**

**Graphical user interface, text

Description automatically generated**

**Description of Figure 1:**

In test 1 when running the script, the input was not manipulated. The expected output would be a linear line. When testing it the outcome was the expected.

**Figure 2: Test 2**

**Graphical user interface, text

Description automatically generated**

**Description of Figure 2:**

In test 2 the input was manipulated by putting my finger on top of the temperature sensor. The expected outcome was a line increasing since the temperature would be increasing. The outcome was quite different because the temperature sensor read the temperature different this is a problem in terms of how the hardware reacts. The temperature sensor is not precise when using my finger. Overall, we notice in the graph that the temperature increases.

**Figure 3: Test 3**

**Graphical user interface

Description automatically generated**

**Description of Figure 3:**

In test 3 the input was manipulated by putting an ice pack close to the sensor. The expected output was a line decreasing since the temperature is decreasing as well. The outcome was as expected there was a line decreasing in temperature.

**\*In all tests we verified that the kill switch works. The code stopped when the button was pressed**

* 2.2. Errors Encountered while Testing
  + - Details about semantics or syntax errors discovered through testing
    - Expanding on “Investigation Column” above

No errors encountered while testing.

* + 2.3. Details and Facts from Testing to Remember
    - Details about concepts that you learned from testing
    - Technical details about interpreter, Arduino equipment, engi1020 module related to testing
    - Anything else you think will demonstrate your learning and will be useful to remember for future

I got more experience identifying and using global and local variables. Also, grew in my understanding of functions and how they work. Something that I learned is the same that I mentioned from implementation. I learned how to plot from the data I collect and how to label the axis of the graph. Also, I could see in this lab how the dot notation and star notation from importing modules works best.

* 3. Reflection and Conclusion
  + 3.1. Reflection Question Answers
    - **Describe in a paragraph or two what you feel to be the most valuable learning experience from our labs.**

The learning gained from labs is enormous. I believe they have enhanced my learning. It’s incredible to see what we learn in class is applied in each lab. Specially since we have hands on experience implementing and testing codes in labs. The satisfaction to see how everything comes together is incredible. The most valuable learning experience from labs is trying to build the code while overcoming the syntax errors and not expected outputs. This type of experience makes me understand concepts better since it makes me visualize that all the learning is related. This also helps me see the millions of applications that coding has now days. I loved doing labs it was a create experience and learned tremendously. The last labs have helped me more since I have been struggling with one of the topics last seen in class. The labs helped me grow my understanding on how functions work. Seeing how global and local variables work with the lab codes have answered lots of questions.

* + 3.2 Additional conclusion Notes