

Lab 2. Introduction to C (Spring 2021)

All students do Lab 2 by themselves (no partner for Lab 2)

[Preparation](#)

[Purpose](#)

[Requirements](#)

[Procedure](#)

[Demonstration](#)

[Deliverables](#)

Preparation

Read Sections 1.12, 2.1, 2.2, 2.3, 2.5, and 2.7 of the book

Find the starter project Lab2_EE319K (EE319K installer, or Canvas directions).

Purpose

The general purpose of this laboratory is to familiarize you with the software development in the C programming language. We choose a problem that exercises problem-solving skills you acquired in EE306 that allow you to devise a solution (algorithm). However, you will code the solution in C instead of assembly.

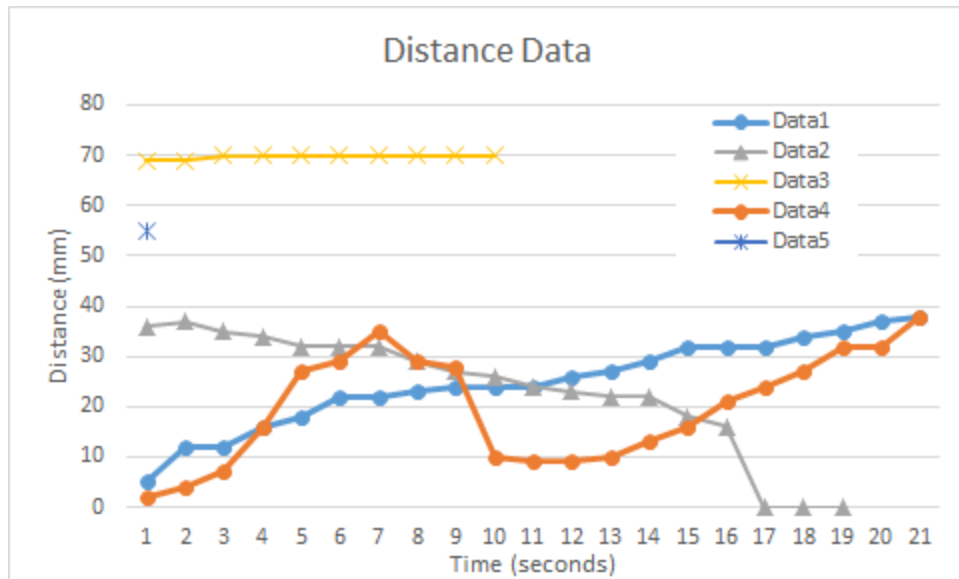
Requirements

The objective of this lab is to write three support routines in **Lab2.c** that are called by a controller in **main.c** to perform data analysis. The controller is collecting distance sensor data periodically. Your task is to write three data analysis routines so the controller can call them as part of its control algorithm. The data exists as an array within the **main.c** program (which you are not allowed to edit), and a pointer to the 16-bit unsigned array is passed to your functions as an input parameter (e.g., `uint16_t Data[]`). The size of the array is passed as a value (e.g., `N`). You may assume the size of the array is less than 50. When `const` is used like this in the parameter declaration, it means the parameter is read-only (your routine can not change its values or its size).

The specific routines are:

1. `uint16_t Average(const uint16_t Data[], const uint32_t N)` - Computes the mean (average) of the distance sensor data collected in the `Data` array. The return result is an integer, therefore any fractional component of the computed mean must be truncated. For this function you may assume there is at least one data value in the array.
2. `int16_t FtoC(const int16_t TinF)` - Convert the temperature in Fahrenheit to the temperature in Centigrade. Assume the input varies from -459 to 1832F. Therefore, the output will range from -273 C to 1000 C. The correct solution uses integer math. You may either truncate or round your result. For example, 224F should convert to 107 C (106.67). However, in integer math, $((224-32)*5)/9$ will only be 106. The automatic grader will allow 106 or 107 or 108 for this lab.
3. `int IsMonotonic(const uint16_t Data[], const uint32_t N)` - Checks whether the recorded readings are an increasing monotonic series (more specifically a non-decreasing series). The controller performs some remedial operation and the desired effect of the operation is to lower the the

distance of the sensed system. If all the values are equal, the result should be true. The following figure shows the test data. Data1 Data3 and Data5 are true. Data 2 and Data4 are false.



Procedure

The starter project provided (Lab2_EE319K) has one assembly file Startup.s and multiple C files, main.c, PLL.c, UART.c, and Lab2.c. All your tasks are performed by writing code for the three subroutines (called functions in C) whose blank stubs are provided in Lab2.c. To test whether your implementations of these functions are correct, you can run the project in the simulator and you should get the following result on the UART window (you will need to enter your name and EID in Lab2.c):

```
UART #1
EE319K Spring 2021 Lab 2
Sensor Data Analysis
Jonathan Valvano EID=jwv123
Test of your Average...ok
Test of your FtoC...ok
Test of your IsMonotonic...ok
Passed all tests - End of Analysis
```

Demonstration

There are [grading sheets](#) for every lab so you know exactly how you will be evaluated. During the demonstration, you will be asked to run your program to verify proper operation. You should be able to single step your program and explain what your program is doing and why. You need to know how to set and clear breakpoints, watch variables like the Readings array and any local variables you declare in your subroutines.

Please make note of which TA checked you out. The name of the TA will greatly help you when resolving any grading issues later.

Do all these well in advance of your checkout

1. **Signup for a Zoom time with a TA. All students do Lab 2 by themselves**
2. **If you cannot live-stream video, create a 60-sec YouTube video and upload it**
3. **Upload your software to canvas, make sure your name is on your software**
4. **Upload your one pdf with deliverables to Canvas**

Do all these during Zoom meeting

1. **Have your one pdf with deliverables open on your computer so it can be shared**
2. **Have Keil Lab 2 open so TA can ask about your code**
3. **Start promptly, because we are on a schedule.**
4. **Demonstrate lab to TA (YouTube video or livestream video)**
5. **Answer questions from TA in the usual manner**
6. **TA tells you your score (later the TA will upload scores to Canvas)**

Deliverables

Upload your Lab2.c file to Canvas. Combine the following components into one pdf file and upload this file also to Canvas. Have the pdf file and Keil open on the computer during demonstration

0. Your name, professor, and EID.
1. A screenshot of your UART1 window showing the final output.

Optional Feedback : <http://goo.gl/forms/rBsP9NTxSy>