Lab 8 grading sheet Circle professor

1) Name Last_____ First_____ EID_____ AC, VT, RY, JV

2) Name Last_____ First_____ EID_____ AC, VT, RY, JV

Use same spelling as listed on Canvas

1. Deliverables 20%:

Submit your ADC.c and Lab8.c as two individual C files 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 names, professors, and EIDs.
1) Circuit diagram showing the position sensor, hand-drawn or PCB Artist, like Figure 8.1 (part a),
2) Four time measurements showing the ADC/LCD execution time (part d)
3) Calibration data, like Table 8.1 (part d)
4) Observations about the Nyquist Theorem (part g)
5) Observations about the Central Limit Theorem (EE319H) (part h)
6) A photo or screenshot verifying the sampling rate is 10 Hz, like Figure 8.8 (part i)
7) Accuracy data and accuracy calculation, Table 8.2 (part j))

2. Performance 35%:
Does it handle correctly all situations as specified?
How pretty is the software?

3. Adhere to coding standard 5%:
Good Names have meaning, Variables have units in comments
Consistent indentation, Consistent use of braces, C99 style

	1)	2)

4. Demonstration 40% (TAs will ask similar, but not exactly identical questions):
You will show the TA your program operation on the actual TM4C123 board. The TA may look at your data and expect you to understand how the data was collected and how the ADC and interrupts work. You should be able to explain how the potentiometer converts distance into resistance, and how the circuit converts resistance into voltage. Also be prepared to explain how your software works and to discuss other ways the problem could have been solved. What will you change in your program if the potentiometer were to be connected to a different ADC pin? How would this system be different if the units of measurement were inches instead of cm? What's your sampling rate? What do you mean by sampling rate? What is the ADC range, resolution and precision? How do you initialize SysTick interrupt? How can you change your sampling rate? Be prepared to prove what the sampling rate is using a calculator and the manual. Explain how, when an interrupt occurs, control reaches the interrupt service routine. Why is it extremely poor style to output the converted data to the LCD inside the SysTick ISR? Where is the interrupt vector located? What are the differences in assembly code generated by the compiler between an interrupt ISR and a regular function? What will happen if you increase your sampling rate a lot? At what point do you think your program will crash? What is the Nyquist Theorem? How does it apply to this lab?

	1)	2)
Total:		

There will be a 10 points penalty if you do not use your LCD code from the previous lab.