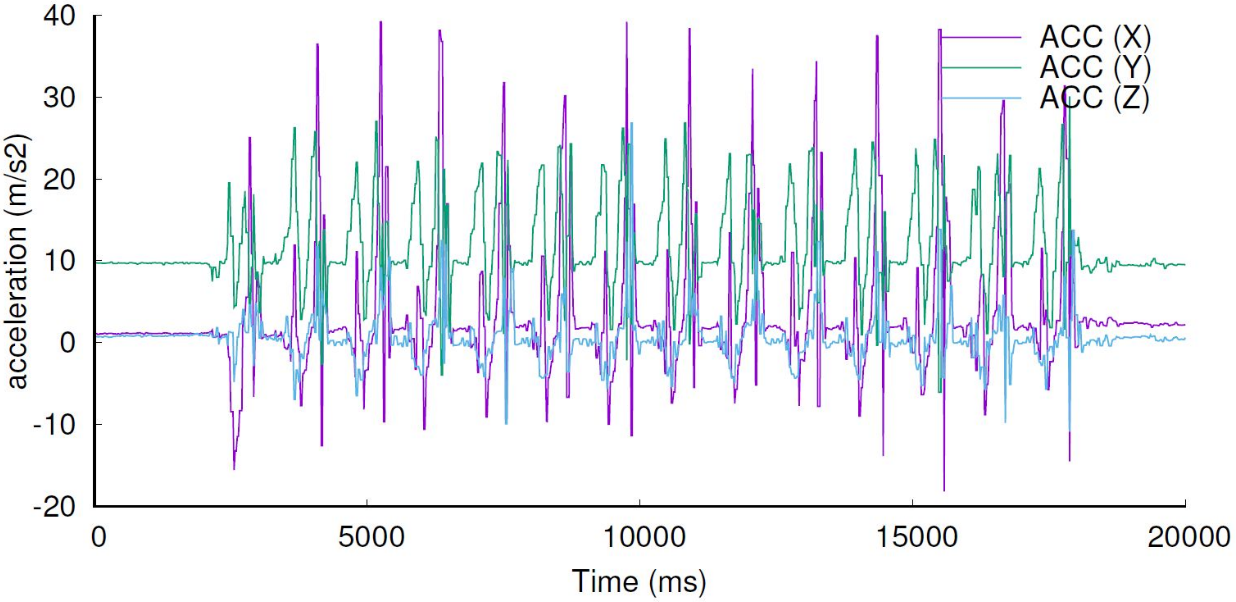
Programming Assignment - 2
Due: 11/17

In this assignment, we will take a look at real smartphone sensor traces (accelerometer, gyroscope and magnetometer/heading).

You have been provided with 3 trace files - **1. Walk-shoe.csv, 2. Walk-pocket.csv and 3. Mag-gyro.csv.** The trace files are in CSV (Comma Separated Values) format. You can open them in Microsoft Office Excel or any other text editor. The first row in the trace file shows the column names. You can use any programming language of your choice to analyze these trace files. The first column in all three test files is time.

In order to understand how the sensor values change over time, it is recommended that you plot the time on X-axis and sensor values on Y-axis. This visualization of data will help you in analyzing the traces. For example, the following plot shows accelerometer data over time for the first trace file “walk-shoe.csv”.



Part 1: STEP COUNTING

In this assignment, you will be writing a program to count the number of steps a user is walking in two different traces (1) walk-shoe.csv and (2) walk-pocket.csv. In walk-shoe.csv, the sensor data is collected from a smartphone that was tied to user’s shoe. In walk-pocket.csv, the sensor data is collected when the smartphone was in user’s pant pocket. For calculating the steps, you can choose to implement any algorithm (peak detection and counting, mean-crossing, autocorrelation or any other) that we discussed in the class. Remember to apply smoothing on the sensor data as an important pre-processing step. Your code should open a trace file, read the sensor values, pre-process it and count the number of step and output it.

You are required to submit the following

1. [15 points] P1 - source code for counting steps in walk-shoe.csv.
    - The output after running P1 should be the correct number of steps
    - Do not submit executable files

2. [5 points] README-P1 - compiling and running instructions for P1
    - Your code should properly compile and run

3. [5 points] P2 - source code for counting steps in walk-pocket.csv.
    - The output after running P2 should be the correct number of steps
    - Do not submit executable files

4. [5 points] README-P2 - compiling and running instructions for P2
    - Your code should properly compile and run

5. [20 points] ANSWERS - in this document, answer the following questions
    - Describe in details what approach/algorithm have you implemented for step counting. Refer to P1 and P2 source codes and explain how your code implements your algorithm. For example, if you are using peak detection algorithm, point to segments of codes in P1 and P2 showing its implementation
    - Explain what changes (if any) you had to make in your algorithm/implementation for counting steps in “walk-shoe.csv” and “walk-pocket.csv”

Part 2: HEADING DETECTION

Open the trace file “mag-gyro.csv”. The tracefile contains data values of 3 gyroscope axis and heading direction (compared to North pole) calculated from magnetometer and accelerometer. During this trace, the user holds the smartphone in her hand and walks around while taking turns. Answer the following questions through analyzing the traces (plotting of the data highly recommended).

6. [10 points]
    a. How many turns does the user take during the walk? How do you determine if there is a turn using the trace?
    b. List the exact times when the turn event happen.

7. [15 points]
    a. Are there any instances of magnetic interference in the trace?

b. If so, how many?
c. How do you figure out that there is magnetic interference using the trace?
d. List the exact times when the magnetic interference events happen.

Grade/points:

75 points for regular questions [10% of the grade]