

Name: _____ Section: _____

Exam3 – Part1

As usual, no credit given to something not taught. I prefer simplicity, efficiency, and least LOC. **NO LOOP. NO IF.**

In the end, submit a folder named Exam3. This folder should contain all the excel sheets, the 1 main code and the 1 function file. *CAUTION*: typically, an error will occur if you try to zip an excel sheet that is currently open. Close all Excel files before zipping.

The data files provided on Canvas contain GPS tracking data of my drive to Virginia during Thanksgiving. Each excel sheet contains information like latitude, longitude, speed, etc... for each state. However, the units (meters and meters per second) are not friendly to represent graphically, and some rows are invalid blank coordinates! Part1 of this exam should extract data that is needed and convert it to more understandable units. Part2 (which you will do Wednesday) will plot the data somehow. During the weekend, feel free to ponder/imagine/guess what I could possibly ask! Code anything! Who knows, you might get it right and have a head start! At worst, you guessed wrong... then delete!! 😊

Specifications/Requirements for part 1 are as follows:

A. The main code:

(3pts) will show a welcome message using a message box. Make sure the program waits for the user to close it before the program continues (3pts). Let the user pick a filename by opening a browser as done in labs (3pts). Only the Excel sheets must show in the browser (3pts). Finally, call the function described below (7pts) to extract and convert data. Omit the semi-colon on this line only so when I run, I see all the variables pop up (2pts).

B. The function:

(5pts) has eight (8) return values:

- A numerical vector with all distances, converted to miles (1m = 0.000621371miles)
- A numerical vector with all speeds, converted to miles per hour (1m/s = 2.23694mph)
- A numerical vector with all elevations, converted to feet (1m = 3.28084feet)
- A numerical vector with the valid latitudes only, another vector with the valid longitudes only
- A char variable: the 2 letter state (SC, FL, NC, etc..)
- and 2 scalars: the average speed in miles per hour, and the number of invalid rows

(3pts) has a valid name (that follows our usual conventions)

(5pts) has one (1) parameter: a filename

(6pts) must have a complete documentation.

Requirements/Specifications:

(10pts) Load the data from the file properly.

(25pts) Slice/extract/convert to correct units properly. For the latitude and longitudes, only take the rows that are valid. For the other data, keep all the rows.

(10pts) To determine the 2 letter state:

For full credit: extract the appropriate letters directly from the filename

For half credit (and so you don't get stuck), I've also put it in the file but be careful of the final data type.

(5pts) Calculate the average speed.

C. Overall

As always, the main code and function should have the usual basics. Use any of the labs as a checklist. (20pts)

(5pts) What was my average speed in each state? _____