**ENGR 102**

**Lab Assignment #2** [**100 Points**]

**Activity #1:** To do in lab – as a team [**50 Points**]

This activity is meant to help illustrate the process of breaking more complex processes into sequential steps, and some of the choices and assumptions involved in doing so.

1. For about 10 minutes: To begin, each member of your team should **individually** create a sequence of instructions, for you to get from your current location to the interior of the TAMUG library (3010 – Jack K. Williams Library (WILL) <http://www.tamug.edu/directions.html> ). Write your instructions as precisely as possible; “Go to the library” is not a good instruction, while “Turn left”, “Walk straight until you are in front of the statue”, “Stop after you pass the entrance to Parking Lot X,” etc. are reasonable. It might help you to bring up <http://www.tamug.edu/directions.html> for a map of campus.
2. Next: As a team, you should look at each other member’s instructions one-by-one. The team members who did not write the instructions should discuss each set of instructions, specifically looking to answer the questions about whether the instructions are clear, whether they provide sufficient detail, and whether they would get someone to the destination. The person who wrote the instructions should not talk during this time. Spend a couple of minutes per set of instructions.
   1. **BEFORE MOVING ON:** Discuss as a team: Which one of the sets of instructions do you consider the “best”? Why?
3. **DO NOT READ UNTIL TASK 2 ABOVE IS COMPLETE!**

As a team: Discuss together and answer the following questions. Your team should produce a single document with (1) copies of each of the sets of instructions (you will need to share, e.g. by Google Drive, your instructions with each other, and (2) short (a couple of sentences, or lists of no more than 10 things) answers to the following questions:

* 1. Which set of the four sequences of steps did you identify as being the best? Why?
  2. In what ways were the sets of sequences that were produced different?
  3. In what ways were the sets of sequences that were produced the same?
  4. Consider whether your choice of which of these would be the best set of instructions might change depending on the person following them. For example (you may think of other examples), would the best set change if:
     1. The person following them was already very familiar with campus, or had never set foot on campus.
     2. The person following the instructions was using a wheelchair, or the person following the instructions was interested in jogging.
     3. The person was in a rush to get to the library, or the person was a visitor interested in getting the best overall feel of campus while traveling to the library.

Briefly describe whether different sets of instructions might have been better options in other scenarios.

* 1. This was a very open-ended question. What questions might you have asked to begin with in order to better know how your sequential steps should have been written?
     1. The point here is to help you understand the importance of requirements gathering at the first stage of attacking a problem – **make sure you are solving the problem someone needs solved, rather than the one you want to solve**.

Save your document as a PDF file and submit your document on ecampus (one per team).

**Activity #2**: To do in lab as a team [**50 Points**]

The purpose of this activity is to practice writing simple programs that require multiple variables, and to ensure you understand the idea of interpolation. The individual assignment will build on this program.

You are to work together as a team to write a short program that performs linear interpolation. Here is the scenario:

You arrive at a racetrack and observe a car moving around a track at what appears to be a constant speed. You would like to be able to predict where the car is at any point in time. To do this, you take a measurement of how far around the track the vehicle has traveled at two points in time. Assume that the track is marked so that you can determine position very precisely. You note the time of this first position measurement. A short while later (before the vehicle has passed the “starting” point on the track), you take a second measurement for how far around the track the vehicle has traveled, again noting the time.

Now, assume that you’d like to reconstruct the position of the vehicle at any time between the first measurement and the second. Since you assume the vehicle is moving at constant speed, this calculation can be found precisely by linear interpolation.

* As a team, determine what variables you will need to use, and what formula(s) you will need to use to perform this calculation. You should use variables for all of the values that could change.

Now, assume that for your observation, the first measurement was taken 30 seconds after you arrived, and the second was taken 45 seconds after you arrived. At the first measurement, the car was 50 meters past the starting line of the track. At the second measurement, the car was 615 meters past the starting line of the track.

* Write a program that determines, for any time between 30 and 45 seconds, where the car will be on the track (in terms of meters past the starting line). The time to evaluate at should be a variable in your program. The program should print both the time and the position at that time to the screen, with a line describing what is being output. You should test your program at various times and make sure the results seem reasonable.
* For your final program that you turn in, you can assume that you want to know the position at a time 37 seconds after you first arrived. (Next week, we will see how you can read in numbers from a user, but for now, just assume it is a fixed number of seconds.)

Turn in your program on ecampus.

*Optional Challenge:* Assume the racetrack is circular with radius 0.5 kilometers. Notice that every time the car passes the starting point of the track, its “distance” from the starting point gets reset to 0. So, if you go far into the future, say at a point 20 minutes after your arrival time, simple linear interpolation will not work. See if you can modify your code to report distances correctly regardless of the time.

If your team does the challenge, turn in your code, computing distance for times of both 37 seconds and 20 minutes.