**ENGR 102 Sect 508**

**Lab 2a**

**50+10 extra points**

**Reading assignment:**

|  |  |
| --- | --- |
| **Lecture Slides** | **L01- L02** |
| **zyBook chapters 1-2** | **Complete all participation and challenge activities** |

***Attention!!***

***For submission: pdf/word file and all py-files as asked in the assignment. No pictures by the phone – it is impossible to read. You will be allowed to resubmit and reupload HW as many times as you want to within the due date/time, only last submission will be graded. No late submissions.***

***For submission you may use this file as template: rename file including your name. Do not forget to put your name inside of this file as well.***

**Activity 1 [50]** To do in lab as a team

***For this submission use Team Header, include all team members into the list of participants. Submit individually until further notice.***

Activity

*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 rate of 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.

**Task 1.1 [20]**

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.
* In the word file I need you to write your derivations and formulas. You will use it when you start typing your code. Make sure that your formula and derivations are correct. The result of your calculations depends on this.

**Task 1.2 [20]**

Now it is time to code. In the comments of your code put

1. Header of the Team
2. Summary of the problem statement. What does it do?
3. Your derived Formula
4. Use comments to make a nice structure of your code.

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.

* Using your derivations from Task 1.1 write a program that determines where the car will be on the track **for any time (**in terms of meters past the starting line) between **30 and 45 seconds**. 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 (I need to know which one is which). You should test your program at various times and make sure the results seem reasonable. Provide your test outputs.
* Collect all of it into the same word/pdf file you were working on during the Task 1.1 for submission
* In the same word/Pdf file make a conclusion of your numerical experiment. For example, what assumptions did you make?

**Task 1.3 [10]**

* 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 from task 1.3 and complete word/pdf file with Tasks 1.1, 1.2, 1.3 on eCampus.

***Optional Challenge******[10 extra points]:*** 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.

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

**[5 points]** How many loops your car will do in 20 minutes?

**Task 1.1**

Variables and formula used:

Initial time = Xa

Initial position = Ya

Final time = Xb

Final position = Yb

Desired time = x

Desired position = y

y= (Ya+((x-Xa)\*(Yb-Ya)/(Xb-Xa)))

**Final program:**

*# By submitting this assignment, all team members agree to the following:  
# “Aggies do not lie, cheat, or steal, or tolerate those who do”  
# “I have not given or received any unauthorized aid on this assignment”  
#  
# Names: Alexia Perez  
# Bethany Gawalis  
# Nicolas Martinez  
# Sam Lyzzaik  
# Tyler Scataglia  
# Section: 508  
# Assignment: Lab 2a  
# Date: 04-09-2018  
# the displacement equation is d=s\*t***import** numpy  
**from** math **import** \*  
  
*#In this problem we will be calculating the position between two given points  
#at any specific moment in time within a given interval.  
#initial time = Xa*Xa=30  
*#initial position = Ya*Ya=50  
*#final time = Xb*Xb=45  
*#final position = Yb*Yb=615  
*#desired time = x*x=37  
*#desired position = y*y = (Ya+((x-Xa)\*(Yb-Ya)/(Xb-Xa)))  
print(**"At time"**)  
print(x)  
print(**"the car's position will be"**)  
print(y)  
print(**"meters past the starting line of the track."**)

Output:

