Project 8B - Matrix Multiplication

Due 4-1

The goal of this project is as follows. Consider a particle moving according to v=(2sin(3*t)+2) m/s. Determine the acceleration of this particle as a function. Although this would be easy enough to do by hand, you must use the following computational approach designed to give you practice with matrix multiplication.

1. Create a 100x1 matrix that stores evenly spaced samples of velocity as a function of time from t=0 to t=1 seconds. Create another matrix that, when applied to any 100x1 sampling of a function estimates the derivative of this function. To do this, you will first need to create a matrix class. Call your matrix class MatrixYOURFIRSTNAME. Store this is a separate file.

2. Add a print function to your matrix, called print, that prints out the values of the matrix in a reasonable way for small matrices.

3. Add a function, called mult, that takes as inputs two matrices. This function should multiply them together and return a third matrix. Each matrix should be stored as an object of type MatrixYOURFIRSTNAME.

4. In this part, you will create a program that uses SOMEONE ELSE'S MATRIX CLASS to estimate the acceleration. Be sure to first try out their print function and verify that you indeed have correctly created your velocity matrix. If you uncover trouble with your partner's files you must contact them to fix it! Do not fix it yourself. If the problem persists you can switch partners. As many times as you need to.

5. Send the result of a(t) to an Excel file. Graph your result.

If you can't get your own matrix class working, you can still submit your Excel graph that was created using someone else's matrix class. Excel graphs created by any other means are not acceptable. Do not submit an Excel graph unless it was created according these instructions. Place Excel and .cpp and .h files into the P8 dropbox.

Due Monday.