1 Introduction

In this project you will explore the implementation of Gaussian Elimination to solve matrix equations of the form Ax = b.

More specifically you are asked to implement two functions with the following signatures:

```
function x = GaussElim (A, b)
function x = GaussElimPP (A, b)
```

Both of these functions take as input an $n \times n$ matrix, A, and an $n \times 1$ vector, b, and attempt to produce the solution to the matrix equation Ax = b. The first variant implements Gaussian Elimination in a straightforward manner as you might find in a standard textbook. The second variant should implement Gaussian Elimination using partial pivoting and row scaling as we discussed in class.

To implement row scaling you should first look at each row of the matrix A and divide through by the coefficient with the largest absolute magnitude (remember to divide the corresponding element of b as well). You need to do this before you start any elimination steps.

To implement partial pivoting you need to permute the rows of the system such that you end up using the pivot element with the largest absolute magnitude at every elimination stage. Again remember that if you permute the rows of A you must also apply the same permutation to b. You should note that the `abs` function can be used to compute the absolute magnitude of an element and the `max` function can be used to find the index of the largest element in a given array. Use the help facility to investigate these functions and any others you may need. Note that your code will be much simpler and faster if you make use of Matlab's vector features that allow you to pull out rows or columns of a matrix and apply operations like scaling to an entire row with one statement.

As part of this assignment you are being provided with a Matlab data file called Project2.mat. You can load this file into Matlab by typing load Project2. The file contains a series of matrix problems, (A1, b1), (A2, b2) etc. You can use whos to list all of the contents. You should solve each of these problems using first GaussElim and then GaussElimPP. You should also compare the results you get with those you would obtain using Matlabs backslash operator ie `A\b`. Make sure to create a text file of the results results clearly labeling which results are from which algorithm are which and include your comments on that document. Remember that you can always check the solution, x, by simply seeing whether Ax is close to b.

HINT: You may find it convenient to simply concatenate A and b into a single matrix, `[A b]`, and to apply the scaling and row swapping operations to that new matrix.

2 Submission Instructions

You will turn in all of the code for this project on Canvas. along with the text file we requested. All of the files for the assignment should be in a single folder on your computer. You will create a zip archive of that folder by right clicking on the folder and selecting the appropriate action. Once the zip archive has been created you can upload it to Canvas. Note that the Canvas site will only accept files with a .zip extension. You should submit all of the functions and scripts we asked you to write or edit along with all of the functions we provided you with.