Note 1: Only one lab this week!

Note 2: No class Tuesday!

Due date/time Saturday, March 11, 9 pm.

[HW8-1] Make a second version of the program jacobi_test so that rather than reading the matrix in, the matrix is set up in the code itself. For a small matrix you could just define each element. For example, replace the read statements from jacobi_test with these lines:

```c
A[0][0]=2.0;
A[0][1]=3.0;
A[0][2]=0.0;
A[1][0]=3.0;
A[1][1]=2.0;
A[1][2]=0.0;
A[2][0]=0.0;
A[2][1]=0.0;
A[2][2]=7.0;
```

and reproduce the results of [HW7-2].

[HW8-2] For a big matrix you will not want to define each matrix element individually as above in HW8-1. Instead, make a third version of the program jacobi_test which uses a loop to set up the dynamical matrix (see notes on coupled-mass-spring systems) for a system of \( N \) masses. Write your code so that it asks you to enter the number of masses \( N \) and the spring constant \( k \). Note that you will have to be careful with the first and last rows of the matrix since they are different from the others.

[HW8-3] Are the eigenvalues that jacobi prints out sorted, e.g. from lowest to highest? If not, write a code in python or C to sort the eigenvalues.

[HW8-4] Run your code from HW8-2 for \( k = 3 \) and \( N = 32 \). Make a plot of the sorted eigenvalues. Do the same for \( k = 3 \) and \( N = 64 \) and \( k = 3 \) and \( N = 128 \). Comment on how the plot is changing with \( N \). Do you recognize the function you are getting? (We will talk about it on Thursday.)