This lab concerns fitting a polynomial to a number of data points. The most basic version of this operation is fitting a line to two points, a task that is old hat to you. If you have a third point, you can't fit a line to this data, but presumably a quadratic would work. A polynomial that passes through given data points is called a **polynomial interpolant** of the data.

The lab has a companion Jupyter notebook. Follow the instructions below and update the corresponding sections of the notebook as needed. For some problems, you will be asked to attach a hand computation to your final output. To do this, you will make a PDF of your Jupyter notebook and then attach to the end of it scanned PDF pages of your handwritten work.

Exercise 1:
To begin, let's think about that line fitting problem again. Suppose we have two points $p_1 = (x_1, y_1)$ and $p_2 = (x_2, y_2)$ and we would like to find the equation of a line $y = mx + b$ going through those points. Substitute these points into the equation of the line to obtain two equations for the unknowns m and b.

Exercise 2:
Suppose $p_1 = (0.2, 0.7)$ and $p_2 = (0.8, -0.4)$. What are the equations for m and b?

Exercise 3:
Write these equations in matrix form:

$$A \begin{bmatrix} b \\ m \end{bmatrix} = v.$$

Explicitly write down what A and v are.

Exercise 4:
Julia has a built-in facility for solving linear systems of equations expressed in matrix form. Follow the instructions in the notebook to solve the system you wrote down in Exercise 3.

Exercise 5:
Verify the m and b you just computed work by plotting the line $y = mx + b$ as well as the two data points that defined the line. The notebook has more information on this task.

Exercise 6:
Now find the equation of a parabola $y = ax^2 + bx + c$ passing through the points $(-1, 1.5)$, $(3, 32.2)$, $(5, -42.6)$. You must

a) Record, by hand, the system of equations to solve.

b) Convert the system into a matrix system.

c) Solve the system using Julia (record the command you used and the solution).

d) Generate a plot that contains the parabola and the three points. Your x coordinate on your plot should range from $x = -2$ to $x = 6$.

Exercise 7:
If you have 7 data points (x_n, y_n), with all of the x_n's different, these uniquely determine a polynomial of some order. What is the order? Enter your response in the notebook.

Exercise 8:
We're going to want to evaluate polynomials with given coefficients and given points. Write a function `poly_eval` in Julia that receives a vector of polynomial coefficients $c = (c_0, \dots, c_n)$ and a vector of x-values, $x = (x_1, \dots, x_k)$ and returns a vector of polynomial values $(p(x_1), \dots, p(x_k))$.

Follow the instructions in the notebook to write this function and test it.

Exercise 9:
For a general polynomial $p(x) = c_0 + c_1 x + c_2 x^2 + \cdots + c_n x^n$, if you have $n+1$ data points (x_k, y_k) and you want $p(x_k) = y_k$, then you obtain $n+1$ equations:

$$c_0 + c_1 x_k + c_2 x_k^2 + \cdots + c_n x_k^n = y_k$$

Equivalently, the coefficients $c = (c_0, c_1, \dots, c_n)$ satisfy

$$Ac = y$$

where $y = (y_1, \cdots, y_{n+1})$ and where

$$A = \begin{bmatrix} 1 & x_1 & \dots & x_1^{n-1} & x_1^n \\ 1 & x_2 & \dots & x_2^{n-1} & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n+1} & \dots & x_{n+1}^{n-1} & x_{n+1}^n \end{bmatrix}.$$

The matrix A is called a Vandermonde matrix. Your task: Write a function in Julia that receives a single vector $x = (x_1, \dots, x_{n+1})$ and returns the associated Vandermonde matrix. Your experience on the Homework 6 Julia problem writing Toeplitz matrices will be helpful! Do that first, if you have not done so already.

Exercise 10:
Suppose we have data points $(1,4)$, $(2,-1)$, $(4,7)$, $(5,-3)$, $(8,1)$, $(9,-10)$ $(11,3)$. Use your 'vandermonde' function to obtain the associated Vandermonde matrix A. Then use Julia to solve $Ac = y$. Finally, plot the the polynomial along with the data points that were used to determine it.

Exercise 11:
Challenge Problem, not due

Suppose we want to make a polynomial approximation of $\cos(x)$. We know the values of $\cos(x)$ exactly at $x = 0, \pi/6, \pi/4, \pi/3, \pi/2, 2\pi/3, 3\pi/4, 5\pi/6, \pi$. We also know that cosine is an even function, so our polynomial should only involve terms of even order. Use Julia to find coefficients $c_0, \dots, c_8$ such that the polynomial

$$p(x) = c_0 + c_1 x^2 + c_2 x^4 + \cdots + c_8 x^1 6$$

satisfies $p(x) = \cos(x)$ for each value of x in the list above. Then generate two plots. The first shows $p(x)$ and the data points it interpolates. The second shows the error $|p(x) - \cos(x)|$ over the range $0 \le x \le \pi$. What is the maximum error?