Due Wednesday, May 28 by the end of class.

Notes:

- The program `lagrange_interp.m` should be used for computing the Lagrange interpolant values in Question 1.
- The program `cheb_points.m` computes the roots of the Chebyshev polynomial of degree n.

1. **[12 points]** In this question we consider interpolating the function $f(x) = \exp\left(\frac{x}{2}\right)$ on the interval $x \in [0, 8]$, using the Lagrange polynomial.

   (a) Suppose we use five equally spaced interpolating nodes at $x = \{0, 2, 4, 6, 8\}$. Derive a theoretical bound on the error in the Lagrange approximation at the following two points:
   (i) $x = 3$ (ii) $x = 7$
   Note: The `prod` command in Matlab may be useful for computing the value of the polynomial component of the error term, $\psi_{n+1}(x)$.

   (b) Use the provided code to compute the actual error in the Lagrange approximation at these two points, and verify that it is within the predicted bounds.

   (c) Repeat part (a) using five Chebyshev points on the interval. (Hint: You will need to map the points provided by `cheb_points.m` from $[-1, 1]$ to $[0, 8]$).

   (d) Repeat part (b) using the same five Chebyshev points.

   (e) Compare the results of part (b) and (d). Does using Chebyshev points as the interpolation nodes give us a better approximation to f at both approximation points? If not, then what is the advantage of using Chebyshev points?

2. **[8 points]**

   (a) Find the values of b_0, b_1, d_0 and d_1 such that the piecewise function

$$S(x) = \begin{cases} S_0(x) = 1 + b_0(x-1) + d_0(x-1)^3 & 1 \le x \le 2 \\ S_1(x) = 1 + b_1(x-2) - \frac{3}{4}(x-2)^2 + d_1(x-2)^3 & 2 \le x \le 3 \end{cases}$$

   defines the cubic spline passing through the points (1,1), (2,1) and (3,0), with **natural** boundary conditions.

   (b) Use the `spline` command in Matlab with the default options to compute a cubic spline passing through the same three points, at the values `xx=1:0.1:3`. Plot this spline along with the function $S(x)$ from part (a) (computed at the same `xx` values) on the same axes. Why are the two splines different?
   Note: logical vectors can be useful to define piecewise functions in Matlab.
   For instance, `(xx >= 1 & xx <= 2)` gives a vector of the same size as `xx` that is equal to 1 for any values between 1 and 2 (inclusive), and 0 otherwise.