Name: ______________________________

Date: ______________________________

Collaborators: ____

Question:	1	2	3	Total
Points:	10	50	10	70
Score:				

Interpolation

1. Determine the interpolating polynomial, $P(x)$, that passes through the points $(0,0)$, $(\frac{1}{2}, \frac{3}{2})$, $(1,1)$.

   (a) (8 points) Construct the Lagrange polynomials $L_i(x)$, $i = 1, \ldots$

   (b) (2 points) Write down $P(x)$:

2. This problem is related to the theory of waves in a medium with variable speed of propagation. The particular setting is underwater propagation of sound.

   The speed of sound in ocean water depends on pressure, temperature, and salinity, all of which vary with depth in a complicated way. Let z denote the depth (in some arbitrary units) under the ocean surface (positive z axis points down) and let $c(z)$ denote the speed of sound at depth z. It is possible to measure $c(z)$ at discrete values of z. The following table is typical of those measurements.

z	$c(z)$
0	5042
500	4995
1000	4948
1500	4887
2000	4868
2500	4863
3000	4865
3500	4869
4000	4874
5000	4879
6000	4887
7000	4905
8000	4918
9000	4933

   A sound speed minimum occurs at a certain depth. The presence of this minimum creates a condition known as *Deep Sound Channel*, permitting guided propagation of underwater sound for thousands of kilometers without interaction with the sea surface or the seabed. The coordinate of the minimum, that is the solution of the equation $c'(z) = 0$, is the depth of the sound channel.

   (a) (10 points) Why it is not a good idea to interpolate the measurement data by a polynomial, $p_n(z)$ (of the degree $n = 13$ in this particular case), and then solve the equation $p'(z) = 0$?

   Plot a graph of the data and its polynomial interpolation. Use the function `polyinterp(x,y,u)` that we developed in class. Store the commands in the matlab script, e.g. **hw07p1.m**. Write your explanation in the README file of your project.

   (b) (10 points) That bring us to spline interpolation. Modify the code of the function `natspline(x,y,u)` that we wrote in class by splitting it into two separate functions: (a) `natspl(x,y)` that returns the arrays b, c, and d of the coefficients of the cubic spline, and (b) `natsplfun(u,x,y,b,c,d)` that calculates the value of the spline at u.

Plot the graph of the spline interpolation of the Runge function on the interval $[-1,1]$. Use $n = 20$ points. Store the commands in the matlab script, e.g. **hw07p2.m**

(c) (10 points) Write a matlab function, `nasplfunp(u,x,y,b,c,d)` that is modeled after `natsplfun(x,y,b,c,d,u)` but returns the value of the interpolation of the derivative of $y(x)$.

(d) (10 points) Modify the function `mybisect(f, left, right)`, so it accepts the vectors x,y,b,c,d as additional arguments (e.g `mybisect2(f,left,right,x,y,b,c,d)`) and uses `natsplfun(x,y,b,c,d,u)`.

(e) (10 points) Write a matlab script (call it e.g. **hw07p3.m**) that finds the depth of the deep sound channel for the data in the table above. Use the full depth span as the initial interval in your `mybisect2`.

Gitlab

3. (10 points) Create a gitlab project called **hw07** (name it exactly as shown). Upload all required matlab code and your readme file; share the project with the instructor.