Name: ______________________________

Date: ______________________________

Collaborators: ____

(Collaborators submit their individually written assignments together)

Question:	1	2	3	4	Total
Points:	30	25	25	10	90
Score:					

Instructor/grader comments:

Interpolation

1. Determine the interpolating polynomial, $P(x)$, that passes through the points $(0,1)$, $(1,4)$, $(2,9)$.

   (a) (15 points) Construct the Lagrange polynomials $L_i(x)$, $i = 1, \ldots$ Write your derivation in the space below.

   (b) (5 points) Write down the interpolating polynomial $P(x)$:

   (c) (10 points) Write matlab function `hw08p1poly(x)` that implements your interpolating polynomial $P(x)$ for a vector argument x. On the same figure plot the three given nodes (using disconnected symbols) and the graph of the polynomial using 100 equidistant point. Provide labels, title, grid.

2. The purpose of the this exercise is to write a script, **hw08p2**, that measures the performance of matlab code implemented in `polyinterp` function. The initial assumption is that the Lagrange interpolation needs $\sim n^2$ operations.

   (a) (15 points) Let's `nmin = 1000` (you may adjust the value of `nmin` so that the running time of your scrip is 30–60 seconds), `nrange = 500`, `nmax = nmin +`

`nrange`, and `nstep = 10`. Preallocate one-dimensional array of the size `(nrange/nstep + 1)` for storing the timing.

For `n = nmin:nstep:nmax` repeat the following steps:

1. evaluate the function `f(x) = abs(x) + x/2 - x`^{2} at n points with the coordinates $x = \cos(\pi m/(n-1))$, $m = 0, \ldots, n-1$.
2. warm up by interpolating of the data using function `vanderinterp` and evaluating the interpolant at a single point $x = 0$.
3. repeat the step again measuring the time for the function call and store the time into the `((n - nmin)/nstep + 1)`th element of the array you preallocated earlier

(b) (5 points) On the same figure plot the graph of time vs. number of nodes and the graph of the function $(n/nmax)^2$. Chose the type of the axis (linear or log). Provide labels, title, grid. Use 'disconnected dots' line style for the measured data graph.

(c) (5 points) Describe your results and conclusions in your README.md file.

3. 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 $s(z)$ denote the speed of sound at depth z. It is possible to measure $s(z)$ at discrete values of z. The following table is typical of those measurements.

z	$s(z)$
0	5042
500	4995
1000	4948
1500	4887
2000	4868
3000	4865
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 $\frac{ds}{dz} = 0$, is the depth of the sound channel. This depth need to be determined with the precision much higher than the separation between the grid points of the measurements.

(a) (10 points) Write a matlab function, `nasplfunp(x,y,b,c,d,u)` that is modeled after `natsplfun(x,y,b,c,d,u)` but returns the value of the derivative of $y(x)$. Recall that for a cubic polynomial $bx + cx^2 + dx^3$ the derivative is $b + 2cx + 3dx^2$.

(b) (15 points) Write a matlab script (call it e.g. **hw08p3.m**) that finds the depth of the deep sound channel, z_{ch} for the data in the table above and the corresponding sound speed s_{min}. Use the full depth span as the initial interval for the `bisect` function.

The values of `z` and `s` are available by using the matlab function `hw08p3data()`.

For solving of the nonlinear equation, define the function

```
fun = @(x) nasplfunp(z,s,b,c,d,x);
```

On the same graph plot the experimental data from the table above (as disconnected symbols), the graph of the function $s(z)$ using the interpolated values at 100 equidistant points, and the point $(z_{\mathrm{ch}}, s_{min})$. Provide the grid, axis labels and the title.

Gitlab

4. (10 points) Create a gitlab project called **hw08** (name it exactly as shown). Upload **all** matlab files that are required to run your code. Share the project with the instructor and the TA and grant them **Reporter** privileges.