MATH 573 ASSIGNMENT 1

Before beginning this assignment, copy the file *Interppoly.m* to the directory you will be using when you start up Matlab.

The first two problems in this assignment study the effect of the choice of interpolation points on the error in polynomial interpolation. These problems should be done in Matlab and you should **hand in an edited output of your Matlab Command Window, along with a printout of any plots you were asked to make**. This edited output should contain the commands you typed in and Matlab's output (when relevant to the problem) along with any explanations you are asked to provide. Make sure you have deleted from your output all errors you have made and any output from Matlab that is not directly related to the problem.

To do this, begin your Matlab session by typing `diary filename` where filename is the name of the file in which you want to store the Matlab output. Now that your diary file is open, enter the command `format compact` (so that when you print out your diary file, it will not have unnecessary spaces) and then the comment line

```
% Math 573 Assignment #1 yourname here
```

Put labels to mark the beginning of you work on each part of each question, so that your edited lab write-up has the format

```
% Question 1(a)
.....
% Question 1(b)
```

and so on. Be sure to answer all the questions in the Matlab portion of the assignment in your Matlab worksheet.

When you finish the assignment, type `diary off` You can now edit and print the file `filename` in the same manner you edit and print any file. This gives an easy way of inserting answers to questions on the assignment that are not directly part of Matlab output. Note that the `print` command is used to place the result of the plot command in a postscript file so you can print it and turn it in. An alternative is to print the plot directly from the graphics window by selecting **File** and then **Print**.

Since the assignment does not assume any prior experience with Matlab, examples of all the commands you will need are provided. If you want to learn more about these commands, select **Help** at the top of the Matlab screen. To save some retyping, it is a good idea to type commands in a file first and then cut and paste them into the Matlab Command Window. Another way to do this is to use a Matlab Script *M-file*. A *script* contains one or more

Matlab commands and is stored as a text file, say `prob1.m` (the extension *.m is required). When you type the name of a script (without the extension *.m) at the Matlab command prompt, the commands within the script are executed. You can also make use of Function *M-files*, which allow you to pass arguments.

If you don’t have a way to access Matlab or have any problems, come to my office for help.

1a. In this problem, we see that taking more and more equally spaced interpolation points sometimes, but not always, produces interpolating polynomials that give better approximations.

Let $[a, b] = [-5, 5]$. For each $n = 4, 8, 16$, and 32, plot over the interval $[a, b]$ the functions $f(x)$ and $P_n(x)$ and then the error function $e(x) = |f(x) - P_n(x)|$, where $f(x) = 1/(1+x^2)$ and $P_n(x)$ is the polynomial of degree $\leq n$ which interpolates f at the points $x_i = a + i(b-a)/n$, $i = 0 \ldots n$. **Record and hand in** the approximate maximum of $|e(x)|$ (as seen from the graph) for each n and approximately where it occurs. This should be done in table of the following form.

Problem 1a		
n	max $\lvert e(x) \rvert$	approx location
4		
8		
16		
32		

To get you started, the case $n = 4$ can be done by typing (or cutting and pasting) the following commands into Matlab

```
diary assign1-output
% Problem 1a
a=-5;  b=5;
n=4;
xin=linspace(a,b,n+1);
yin= 1./(1+xin.^2);
xout=linspace(a,b,100);
[yout, cof] = Interppoly(xin,yin,xout);
ytrue= 1./(1+xout.^2);
plot(xout,yout,xout,ytrue)
title(’Problem 1a functions; n=4’);
print -deps assign1-out1.eps
pause(5);
plot(xout,abs(ytrue-yout))
title(’Problem 1a error; n=4’);
print -deps  assign1-out2.eps
```

The other values of n can be done by by entering a new value of n and then re-entering the other commands (with possibly minor modifications).

1b. Repeat Problem (1a) for the function $g(x) = \exp(-.4x^2)$, which may be evaluated at the vector of points *xout* by typing:

```
g= exp(-.4*xout.^2);
```

2. In this problem, we see that the choice of interpolation points can make a big difference in the quality of the approximation.

Repeat Problem (1a), this time using the polynomial $Q_n(x)$ of degree $\leq n$ which interpolates f at the (Chebyshev) points

$$x_i = \frac{a+b}{2} + \frac{b-a}{2} \cos\left(\frac{(2i+1)\pi}{2n+2}\right), \quad i = 0, \dots, n.$$

This Matlab code will be similar to the one in Problem 1. The main difference is that instead of generating the interpolation points by the command

```
xin=linspace(a,b,n+1);
```

you can use the commands

```
x=linspace(1,2*n+1,n+1);
xin = (a+b)/2 + (b-a)/2 * cos(x*pi/(2*n+2));
```

Be sure to save your output to a different file.

3a. Without solving any equations, find a cubic polynomial $P_3(x)$ satisfying $P_3(0) = 0$, $P_3(1/3) = 0$, $P_3(2/3) = 0$, and $P_3(1) = 1$.

3b. Suppose $P_2(x)$ is a given quadratic polynomial and x_i, $i = 0, 1, 2, 3$ are distinct points. Construct in as simple a manner as possible a cubic polynomial $P_3(x)$ satisfying $P_3(x_i) = P_2(x_i)$, $i = 0, 1, 2$, and $P_3(x_3) = 1$. Your answer will involve $P_2(x)$.

4. Consider the problem of determining a polynomial $P_2(x)$ of degree ≤ 2 satisfying

$$P_2(x_0) = f(x_0), \qquad P_2'(x_1) = f'(x_1), \qquad P_2(x_2) = f(x_2),$$

where f, x_0, x_1, x_2 are given and $x_0 < x_1 < x_2$. Does this problem have a solution for all given f in $C^1[x_0, x_2]$ and all x_0, x_1, x_2 ? If so, find it; if not, give an example for which there is no solution.

Note that this problem does not fit the theory developed in class.