Homework 1
Math 128A
Due Monday, 10/13/08, 11:00 a.m.

1. Consider the function
$$f(x) = \frac{1-\cos(x)}{x^2}.$$
   (a) Analytically evaluate $\lim_{x\to 0} f(x) = L$.
   (b) As $x \to 0$, at what rate does $f(x) \to L$?
   (c) Compute $f(x)$ as written on a computer for values of $x = 10^{-1}, 10^{-2}, \ldots, 10^{-10}$. Comment on your results.
   (d) Suppose that we are able to represent floating point numbers with N decimal digits of accuracy. Around what value of r will the evaluation of $f(x)$ produce very large relative errors when $|x| < r$?
   (e) Rearrange the expression for $f(x)$ to a mathematically equivalent expression so that the this new function evaluates accurately for very small values of x. Verify the success of your rearrangement computationally. Are there values of x where you expect accuracy problems with your rearrangement?

2. The Taylor series about $x = 0$ for the arctangent function converges for $-1 < x \leq 1$ and is given by
$$\tan^{-1}(x) = \sum_{n=0}^{\infty} \frac{(-1)^n x^{2n+1}}{2n+1}.$$
   (a) Write a computer program to evaluate $\tan^{-1}(x)$ by truncating the series to $N+1$ terms.
   (b) Use your program to approximate π by evaluating $\tan^{-1}(1)$ using truncations of $N = 10^1, 10^2, \ldots 10^6$.
   (c) Repeat part (b) using $x = 3^{-1/2}$.
   (d) Plot the errors in the approximations from parts (b) and (c) vs. the number of terms in the truncated sum on a log-log graph and a semilog (log-linear) graph. To compute the errors, use that $\pi \approx 3.141592653589793$. What do these graphs indicate about the rate of convergence?
   (e) Analytically derive a bound for the error for the series in parts (b) and (c).

3. You are running a simulation that updates time every 0.1 seconds. The time in the simulation is kept by incrementing a time variable. See the below code.

```
dt = 0.1;         % time step
t  = 0;           % initialize time
Nsteps = 864000;  % number of steps to take

% loop in time
%
```

```
for j=1:Nsteps

   %
   % SOME SIMULATION
   %

   % update time
   %
   t = t + dt;
end
```

(a) Suppose you are simulating one day (86,400 s). Implement the above code, and compute the absolute and relative errors in the time at the end of the simulation.

(b) Change the time increment to 0.125 and again run the simulation to 1 day. What are the relative and absolute error in the time?

(c) Explain the difference in the results from parts (a) and (b).

4. Suppose you are constructing a table of values over one period of the function

$$f(x) = \mathrm{e}^{\cos(x)}.$$

The x values of the table are to be equally spaced ($h = x_{j+1} - x_j$ for all j). If second degree polynomial interpolation is used to approximate f at arbitrary x values using the three closest table values, how small does h need to be to guarantee that this produces values within an absolute error of 10^{-6}?

5. Neville's method is implemented in the MATLAB program below. Modify the code so that the algorithm terminates when the difference between successive diagonal entries in the Neville table are less than some user supplied ϵ.

```
% Neville's method
% Input:   x - a vector with the nodes
%          y - a vector with the corresponding function values
%         xi - the location to evaluate the interpolating polynomial
%
% Output: yi - the value of the interpolating polynomial at xi
%
function yi = neville(x,y,xi);
  N = length(x);
  p = zeros(N);
  p(1,1) = y(1);

  for i=2:N
    p(i,1) = y(i);

    for j=2:i
```

```
      num = (xi-x(i))*p(i-1,j-1) - (xi-x(i-j+1))*p(i,j-1);
      den = x(i-j+1) - x(i);
      p(i,j) = num/den;
    end
  end
  yi = p(i,i);
```

(a) The function $Ei(x)$ is defined as

$$Ei(x) = \int_x^\infty \frac{e^{-t}}{t}\, dt.$$

Use your modified code to compute the value of $Ei(1.625)$ with $\epsilon = 10^{-6}$. What degree polynomial was used in your approximation?

x	Ei(x)
1.000000000000000	0.219383934395520
1.100000000000000	0.185990904536040
1.200000000000000	0.158408436851462
1.300000000000000	0.135450957849129
1.400000000000000	0.116219312571358
1.500000000000000	0.100019582406633
1.600000000000000	0.086308333697540
1.700000000000000	0.074654644401253
1.800000000000000	0.064713129363869
1.900000000000000	0.056204378174535
2.000000000000000	0.048900510708061

(b) How do the differences in the diagonal elements of the Neville table compare with the actual errors? Use that $Ei(1.625) \approx 0.083216796544033$.

(c) Rearrange the data so that the points are ordered from closest to farthest from the interpolation point. Compare the differences in diagonal elements of the Neville table with this reordering of the data to those produced by using the data is its original ordering.

6. The following table (from J. Keller, *SIAM Review*, vol. 26, 267-268, 1984) gives the probability, q, that a given racquetball player will shutout an opponent as a function of the probability, p, that the player will win any particular rally regardless of who serves. Use Neville's method to approximate q given that $p = 0.6$. What order polynomial would you use to approximate q? Comment on the accuracy of the approximation for interpolating polynomials of different degrees.

p	q
1.00	1.0000000
0.90	0.7530000
0.850	0.5340000
0.842	0.5000000
0.840	0.4900000
0.500	0.0001504