Brian Knisely ME523 Assignment 2

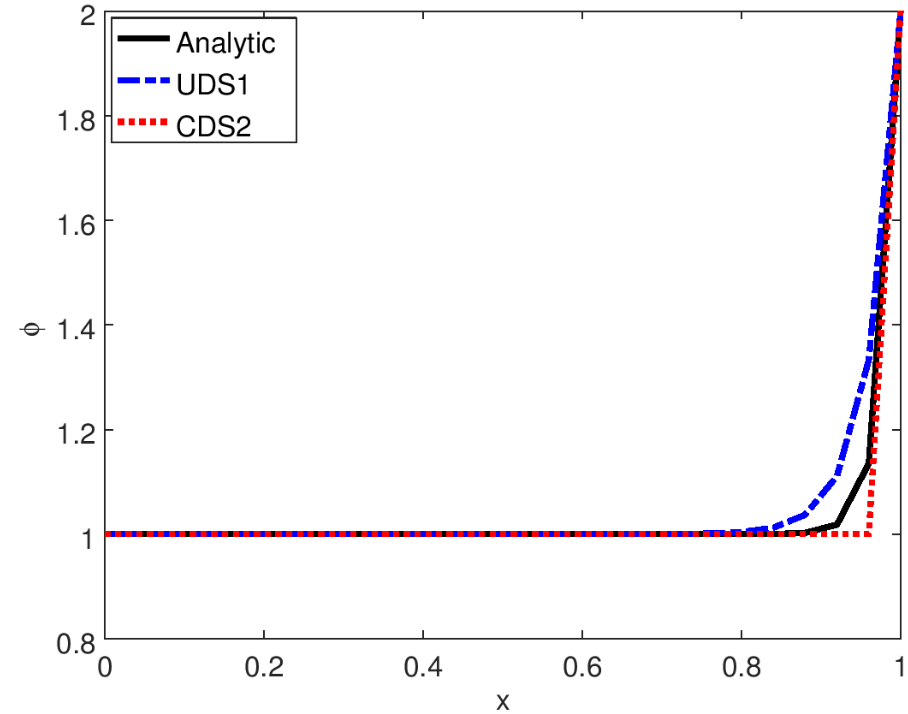
Problem 1.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% compute the numerical solution using CDS convection and CDS diffusion %%%

function numerical_1d_cds
  % compute the numerical solution of the steady, linear, 1D convection-diffusion
  %   problem at each grid point using a finite-difference method
  %   with CDS convection and CDS diffusion

  % Call global variables needed by this function
  global phil phir np1 n den vel dif x
  global aw ap ae q phi

  % global variables set by this function
  global phicds Pe

  % Apply BCs
  phicds(1) = phil;
  phicds(np1) = phir;

  % compute product of density and velocity
  denvel = den*vel;

  % loop over interior grid points (i = 2 to i = n)
  for i = 2:n

      % CDS convection contributions
      awc = -denvel / ( x(i+1) - x(i-1) );  % West coefficient
      aec =  denvel / ( x(i+1) - x(i-1)   );  % East coefficient
      apc = 0;  % Coefficient at point

      % CDS diffusion contributions (unchanged)
      %dxr = 2.*dif / ( x(i) - x(i-1) );  %
      awd = -dif / ( (x(i) - x(i-1))*(x(i+1) - x(i-1))/2 );  % West coefficient
      aed = -dif / ( (x(i+1) - x(i-1))*(x(i+1) - x(i))/2 );  % East coefficient
      apd = 2*dif / ( (x(i) - x(i-1))*(x(i+1) - x(i)) );  % Coefficient at point

      % fill row i coefficient matrix and right-hand-side values
      aw(i) = awc + awd;
      ae(i) = aec + aed;
      ap(i) = apc + apd;
      q(i)  = 0.;
  end


  % take care of the boundary conditions
  i     = 2;
  q(i)  = q(i) - aw(i)*phicds(i-1);
  aw(i) = 0.;

  i     = n;
  q(i)  = q(i) - ae(i)*phicds(i+1);
  ae(i) = 0.;

  phi(1)   = phicds(1);
  phi(np1) = phicds(np1);

  %--- solve for phi using the TDMA direct solver ---

  tdma;
```

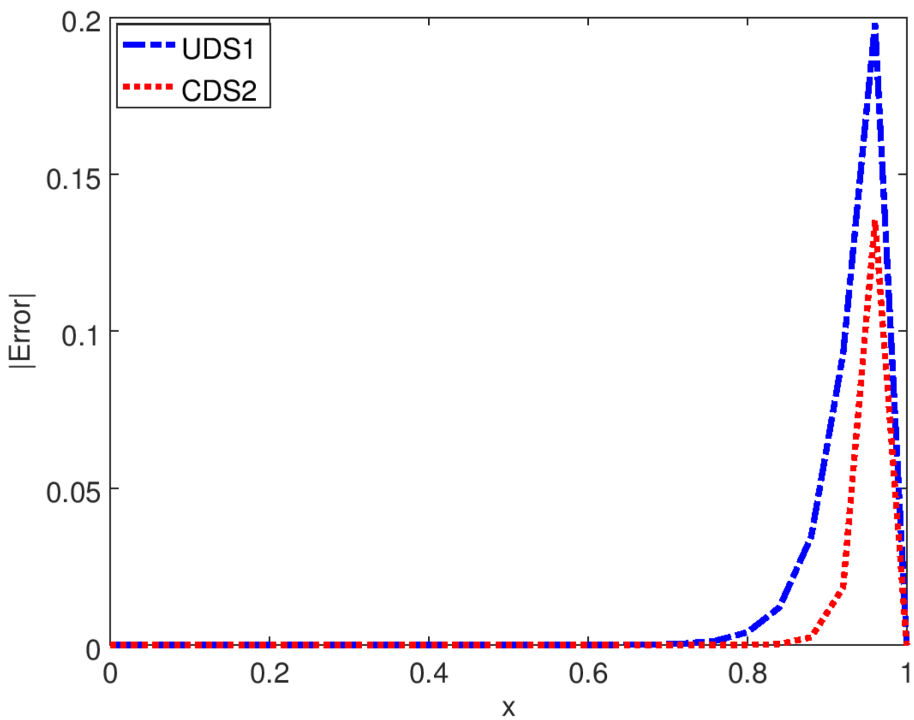
```
  % copy the solution from phi to phicds
  for i=2:n
      phicds(i) = phi(i);
  end

end

%%% end of numerical_1d_cds %%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% compute errors of the numerical solutions wrt/the analytic solution %%%

function errors
  % compute differences between two numerical solutions and the analytic solution

  # global inputs to this function
  global phicds phiuds phian

  # global outputs to this function
  global errcds erruds

  erruds = abs(phian - phiuds);  % compute error in uds solution
  errcds = abs(phian - phicds);  % compute error in cds solution

end

%%% end of errors %%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% generate figures %%%

function figures
  % plot the analytic solution and two numerical solutions as functions of x
  % plot errors between two numerical solutions and the analytic solution
  %   as functions of x
  % save both figures to the current working directory as .png files

  global x phian phiuds phicds erruds errcds

  % Make figure for phi vs x for analytical, UDS, and CDS solutions
  figure(1)  % Create Figure "1"
  % Plot Phi_analytic vs x with black line
  % Plot Phi_UDS vs x with dash-dot blue line
  % Plot Phi_CDS vs x with dotted red line
  p = plot(x, phian, 'k', x, phiuds, 'b-.', x, phicds, 'r:');
  set(p, 'linewidth', 3)  % Set line thickness to 3
  xl = xlabel('x'); yl = ylabel('\phi');  % Add axis labels
  % Add legend and specify its location
  l = legend('Analytic', 'UDS1', 'CDS2', 'location', 'northwest');
  % Increase font size of plot elements
  set([gca, l, xl, yl], 'fontsize', 16);
  figure(1, 'position', [50 50 600 450])
  saveas(1,'hw2_figure1.png');  % Save figure to file "hw2_figure1.png"
  close(1)  % Close the figure

  % Make figure for |error| vs x for UDS and CDS solutions
  figure(2)  % Create Figure "2"
  p = plot(x, erruds, 'b-.', x, errcds, 'r:');
  set(p, 'linewidth', 3)  % Set line thickness to 3
  xl = xlabel('x'); yl = ylabel('|Error|');
  % Add legend and specify its location
  l = legend('UDS1', 'CDS2', 'location', 'northwest');
```

```
  % Increase font size of plot elements
  set([gca, l, xl, yl], 'fontsize', 16);
  figure(2, 'position', [650 50 600 450])
  saveas(2,'hw2_figure2.png');   % Save figure to file "hw2_figure2.png"
  close(2)  % Close the figure

end

%%% end of figures %%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% write an output file %%%

function outfile
% write an ascii text output file
% echoes input parameters and key derived quantitites,
%   and generates six columns of output for each grid point:
%     x phian phiuds phicds erruds errcds

% Call global variables needed to state at top of text file
global xmin xmax den vel dif phil phir n dxrat dxmin dxmax pe pedxmin pedxmax

% Call global variables needed for table
global x phian phiuds phicds erruds errcds

  % Name output file and open it
  outfile = 'hw2_results.txt'
  fid = fopen(outfile, 'w');  % Open a file to store results

  % Print each variable in nice formatting
  fprintf(fid, 'Input quantities:\n')
  fprintf(fid, '%8s = %12.6f\n', 'xmin', xmin)
  fprintf(fid, '%8s = %12.6f\n', 'xmax', xmax)
  fprintf(fid, '%8s = %12.6f\n', 'den', den)
  fprintf(fid, '%8s = %12.6f\n', 'vel', vel)
  fprintf(fid, '%8s = %12.6f\n', 'dif', dif)
  fprintf(fid, '%8s = %12.6f\n', 'phil', phil)
  fprintf(fid, '%8s = %12.6f\n', 'phir', phir)
  fprintf(fid, '%8s = %12.6f\n', 'n', n)
  fprintf(fid, '%8s = %12.6f\n', 'dxrat', dxrat)
  fprintf(fid, '\nDerived quantities:\n')
  fprintf(fid, '%8s = %12.6f\n', 'dxmin', dxmin)
  fprintf(fid, '%8s = %12.6f\n', 'dxmax', dxmax)
  fprintf(fid, '%8s = %12.6f\n', 'pe', pe)
  fprintf(fid, '%8s = %12.6f\n', 'pedxmin', pedxmin)
  fprintf(fid, '%8s = %12.6f\n', 'pedxmax', pedxmax)

  % Print header
  fprintf(fid, '\nTabulated results:\n')
  fprintf(fid, '\n%3s,%12s,%12s,%12s,%12s,%12s\n',...
      'x', 'phian', 'phiuds', 'phicds', 'erruds', 'errcds')

  % Print results one line at a time
  for i = 1:length(phian)
      fprintf(fid, '%3.2f,%12.6e,%12.6e,%12.6e,%12.6e,%12.6e\n',...
      x(i), phian(i), phiuds(i), phicds(i), erruds(i), errcds(i))
  end

  fclose(fid);  % Close the file

end
%%% end of outfile %%%
```

Problem 2.



Analytic
UDS1
CDS2
φ
x
0.8
1
1.2
1.4
1.6
1.8
2
0
0.2
0.4
0.6
0.8
1



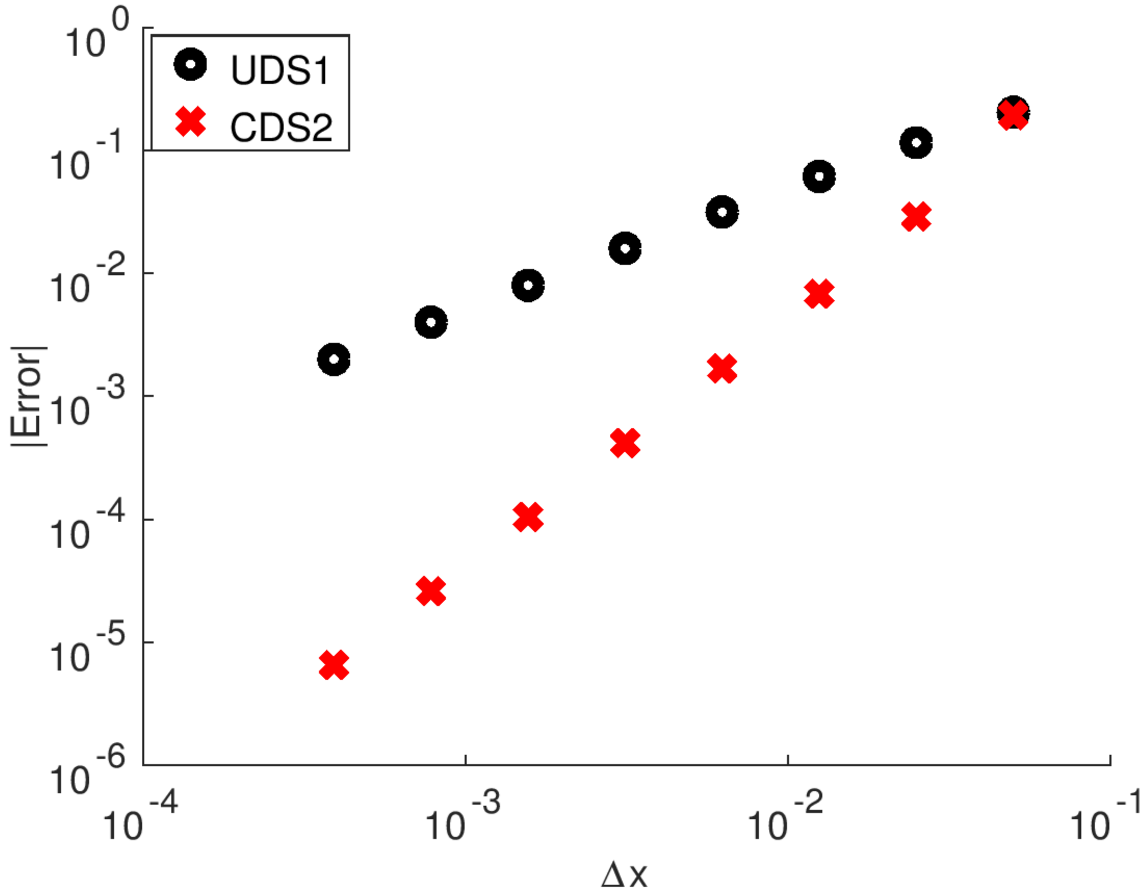
UDS1
CDS2
|Error|
x
0
0.05
0.1
0.15
0.2
0
0.2
0.4
0.6
0.8
1

Output in text file:

```
Input quantities:
    xmin =     0.000000
    xmax =     1.000000
     den =     1.000000
     vel =    10.000000
     dif =     0.200000
    phil =     1.000000
    phir =     2.000000
       n =    25.000000
   dxrat =     1.000000

Derived quantities:
   dxmin =     0.040000
   dxmax =     0.040000
      pe =    50.000000
 pedxmin =     2.000000
 pedxmax =     2.000000

Tabulated results:

  x,       phian,      phiuds,      phicds,      erruds,      errcds
0.00,1.000000e+000,1.000000e+000,1.000000e+000,0.000000e+000,0.000000e+000
0.04,1.000000e+000,1.000000e+000,1.000000e+000,2.360556e-012,0.000000e+000
0.08,1.000000e+000,1.000000e+000,1.000000e+000,9.441781e-012,0.000000e+000
0.12,1.000000e+000,1.000000e+000,1.000000e+000,3.068612e-011,1.110223e-016
0.16,1.000000e+000,1.000000e+000,1.000000e+000,9.441870e-011,1.110223e-016
0.20,1.000000e+000,1.000000e+000,1.000000e+000,2.856166e-010,2.220446e-016
0.24,1.000000e+000,1.000000e+000,1.000000e+000,8.592111e-010,3.330669e-016
0.28,1.000000e+000,1.000000e+000,1.000000e+000,2.579994e-009,8.881784e-016
0.32,1.000000e+000,1.000000e+000,1.000000e+000,7.742342e-009,2.331468e-015
0.36,1.000000e+000,1.000000e+000,1.000000e+000,2.322938e-008,1.321165e-014
0.40,1.000000e+000,1.000000e+000,1.000000e+000,6.969045e-008,9.403589e-014
0.44,1.000000e+000,1.000000e+000,1.000000e+000,2.090733e-007,6.920020e-013
0.48,1.000000e+000,1.000001e+000,1.000000e+000,6.272192e-007,5.109579e-012
0.52,1.000000e+000,1.000002e+000,1.000000e+000,1.881637e-006,3.775147e-011
0.56,1.000000e+000,1.000006e+000,1.000000e+000,5.644749e-006,2.789471e-010
0.60,1.000000e+000,1.000017e+000,1.000000e+000,1.693303e-005,2.061154e-009
0.64,1.000000e+000,1.000051e+000,1.000000e+000,5.079003e-005,1.522998e-008
0.68,1.000000e+000,1.000152e+000,1.000000e+000,1.523033e-004,1.125352e-007
0.72,1.000001e+000,1.000457e+000,1.000000e+000,4.564158e-004,8.315287e-007
0.76,1.000006e+000,1.001372e+000,1.000000e+000,1.365598e-003,6.144212e-006
0.80,1.000045e+000,1.004115e+000,1.000000e+000,4.069826e-003,4.539993e-005
0.84,1.000335e+000,1.012346e+000,1.000000e+000,1.201022e-002,3.354626e-004
0.88,1.002479e+000,1.037037e+000,1.000000e+000,3.455828e-002,2.478752e-003
0.92,1.018316e+000,1.111111e+000,1.000000e+000,9.279547e-002,1.831564e-002
0.96,1.135335e+000,1.333333e+000,1.000000e+000,1.979981e-001,1.353353e-001
1.00,2.000000e+000,2.000000e+000,2.000000e+000,0.000000e+000,0.000000e+000
```

Problem 2 Discussion: The analytical solution resulted in the expected shape for a high Peclet number. Both approximations were able to capture the same shape as the analytical solution within some error. In both approximations, the largest errors were near x = 0.95, where the slope of the analytical solution is steepest. The CDS2 approximation had lower error than the UDS1 at all points, especially near x = 0.95.

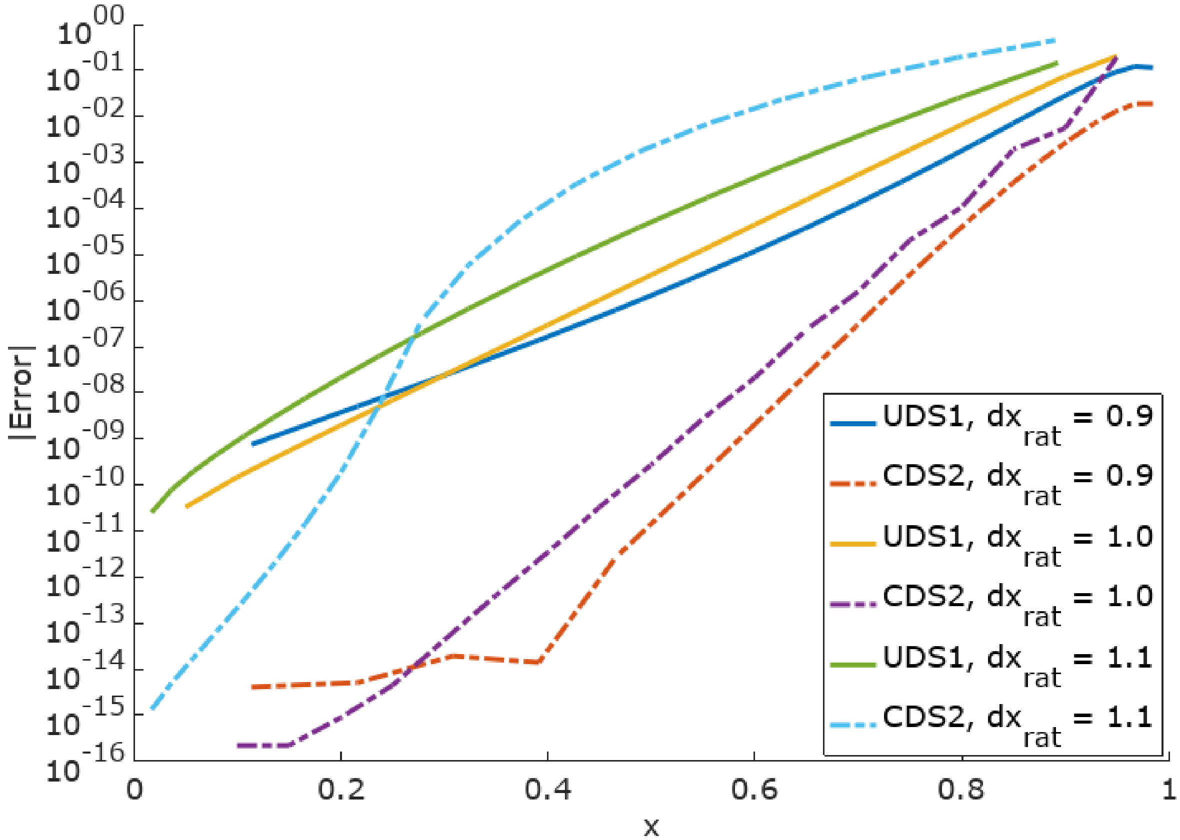
Problem 4.



For UDS1, P = 0.9957
For CDS2, P = 2.0004

Problem 4 Discussion: As expected, the second-order scheme converged more quickly than the first-order scheme. When the P-values were calculated, the first-order scheme had a P-value equal to approximately 1, while the second-order scheme had a P-value equal to approximately 2. As shown in the figure comparing convergence rates, both schemes converged linearly as Δx decreased.

Problem 5.



Problem 5 Discussion: In the analytical curve described previously, the exact solution features steep gradients in the range of x = 0.8 to x = 1.0. For this case, we expect that a more refined grid near the steep gradients would help resolve the solution with reduced error compared to a uniform grid. As shown above, reducing the dxrat parameter, which results in smaller cells near the right-hand side of the figure than on the left-hand side, results in lower error in the regions of high gradients in the solution. The effect was more pronounced with the CDS2 scheme than with the UDS1 scheme, but both benefitted from smaller cells near x = 1.0. Because the number of divisions was held constant across all tests, the increased spacing near x = 0 resulted in higher error for the low dxrat cases compared to the dxrat = 1 case. In every case, the highest errors were located near x = 0.95, where the gradients in the solution are steepest.