ME 481 – H14 **Name ________________________**

1) Copy the MATLAB files "ode_smd_main.m" and "ode_smd_yd.m" below into two separate M files in your working directory. Use the exact file names. Run "ode_smd_main.m" to provide a forward dynamic simulation of a spring-mass-damper. Provide MATLAB plots of position and velocity as a function of time and a MATLAB plot of time step as a function of time. Also provide a phase plane plot of velocity as a function of position.

2) Compute damping ratio ζ based on spring-mass-damper coefficients in file "ode_smd_main.m" and using log-decrement methods from the decay envelope in your simulation data. Show your work.

ζ_{COEFF} __________________ $\zeta_{\text{LOG-DEC}}$ __________________

3) Modify "ode_smd_yd.m" for Coulomb friction force equal to f = 0.5 Newtons and viscous damping c = 0. Use the simple friction model $F_{\text{FRICTION}} = -f * \text{sign}(\dot{x})$. Provide MATLAB plots of position and velocity as a function of time and a MATLAB plot of time step as a function of time. Also provide a phase plane plot of velocity as a function of position. Provide a listing of your code. Comment on the differences from part 1) above. Specifically address time step and CPU time.

4) Compute Coulomb friction force based on the decay envelope in your simulation data from part 3). Show your work

f_{COULOMB} __________________

5) Modify your MATLAB code to simulate the stick-slip drag-sled from H12 part 2). You must use the more detailed friction model on page 1 of Notes_08_06. The simple friction model used above will not suffice. Provide MATLAB plots of position, velocity and <u>acceleration</u> of the drag-sled as a function of time and a MATLAB plot of time step as a function of time. Provide a listing of your code. Comment on the differences from H12.

EXTRA CREDIT
Modify your pendulum from H13 using “Script Pin Friction” with “Pin Radius = 0.25 inch” and “Friction Coefficient = 0.1” for ±10 degrees of motion. Attach a screen shot of your mechanism. Provide a phase plane plot and estimate joint friction torque from the time decay envelope.

ME 481 – H14 **Name** ________________________

```
% ode_smd_main.m - main program for example use of ODE solver
%   spring-mass-damper
% HJSIII, 20.04.09
%
% ODE coded in file ode_smd_yd.m  - m*xdd = Fext + Fspr + Fdamp
%
% {y} = { x  }          {yd} = { xd  }
%       { xd }                 { xdd }

clear

global  m  k  c

% physical constants
% x [m]
% xd [m/sec]
% xdd [m/sec^2]
m = 1;              % mass [kg]
k = 157.9;          % spring [N/m] - causes wn = 2Hz
c = 2;              % viscous [N.sec/m] - causes 4 sec exp decay

% initial conditions
y0 = [ 0.1  0  ]';   % free release

% time range
tspan = [ 0  4 ];

% measure CPU time
tic;
[ t, y ] = ode23( 'ode_smd_yd', tspan, y0 );
t_exe = toc

% time step
h = 1000 * diff(t);   % units [msec]
h = [ h ; h(end) ];   % repeat last value to make the same length as t

n_time_steps = length( t )
ave_time_step = mean( h )

% time domain results
figure( 1 )
  subplot( 2, 2, 1 )
    plot( t, y(:,1) )
    xlabel( 'Time [sec]' )
    ylabel( 'Position [m]' )
    axis( [ 0 4  -0.1 0.1 ] )
  subplot( 2, 2, 2 )
    plot( t, y(:,2) )
    xlabel( 'Time [sec]' )
    ylabel( 'Velocity [m/sec]' )
    axis( [ 0 4  -1.5 1.5 ] )
  subplot( 2, 2, 3 )
    plot( y(:,1), y(:,2) )
    xlabel( 'Position [m]' )
    ylabel( 'Velocity [m/sec]]' )
  subplot( 2, 2, 4 )
    plot( t, h )
    xlabel( 'Time [sec]' )
    ylabel( 'Time step [msec]' )

% bottom - ode_smd_main
```

ME 481 – H14 **Name ________________________**

```
function yd = ode_smd_yd( t, y )
% provides yd for integration
% spring-mass-damper
%    m*xdd = Fext + Fspr + Fdamp
% HJSIII, 20.04.09

global m  k  c

% free motion
Fext = 0;

% individual terms
x = y(1);
xd = y(2);

% ordinary differential equation (ODE)
Fspr = -k*x;
Fdamp = -c*xd;
xdd = ( Fext + Fspr + Fdamp ) / m;

% return values
yd(1,1) = xd;
yd(2,1) = xdd;

return
% bottom - ode_smd_yd
```