All Assignments

Use MATLAB wherever possible to work the problem or check your work on a problem. ***Whenever a requested problem asks you to plot or sketch the answer, you must use MATLAB to do your work***.

Treat the homework like a quiz! In other words, don't do the homework with the notes open. Instead, study and learn the material as well as you can, and then try to work the homework problems. If you get stuck, cover up the homework, re-read the notes, and try again.

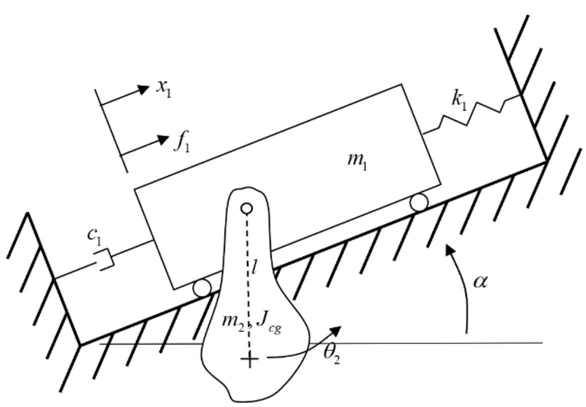
If you work homework as a group, you **<u>must</u>** identify the group*.

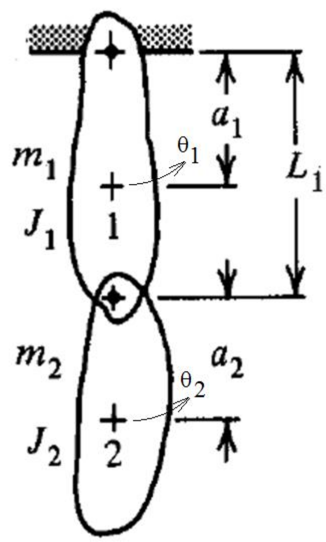
Assignment-10

- Reading
  - Handout sections #11, #13 & #14
- Reference
  - *Tse, Morse, & Hinkle* Chapter 4
- Homework

  For each of the following figures, develop the exact and linearized equations of motion by both Newton's Method and Lagrange's Method. Express the linearized equations of motion in matrix form.

  - **10-A)** *For this problem, the angle α is considered constant* **10-B)**

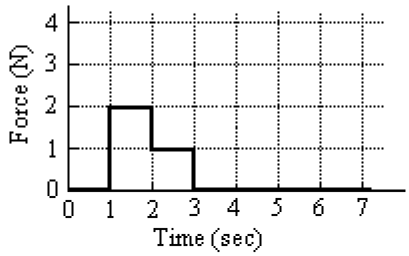


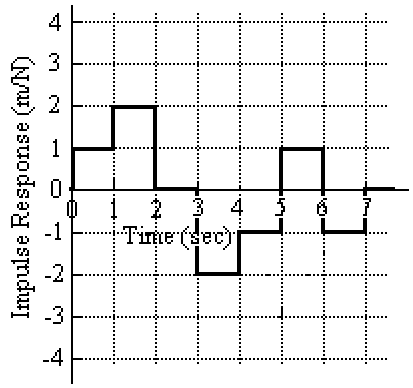


---

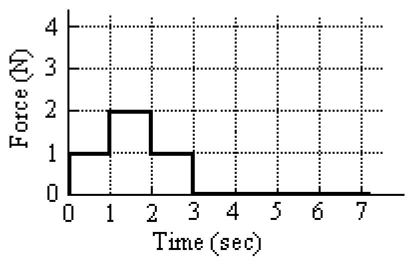
* *Remember that failure to provide proper reference/citation is called* ***<u>plagiarism.</u>***

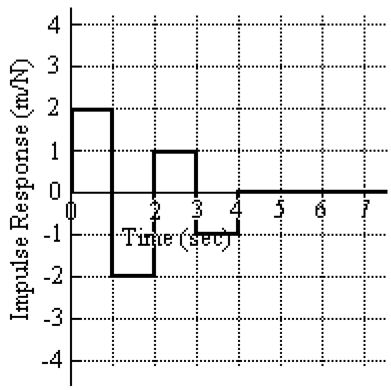
- **10-C)** For the following force and impulse response, evaluate the response of the system at each 1 second increment from 0 to 7 seconds, x(0), x(1), x(2), x(3), x(4), x(5), x(6), and x(7). Show the details of your work (you may wish to use sketches to show the intermediate steps and to aid in the solution).





- **10-D)** Repeat Problem 10-D for the following force and impulse response.





- **10-E)** The single degree-of-freedom simple pendulum (undamped, unforced) has the following *exact* equation of motion.

$$ml^2\ddot{\theta} + mgl\sin\theta = 0$$

To linearize, the approximation, $\sin\theta \approx \theta$, is used. This substitution results in the following *linearized* equation of motion.

$$\ddot{\theta} + \frac{g}{l}\theta = 0$$

The purpose of this assignment is to evaluate the results of the linearizing assumption on the predicted system response. For the first two plots mentioned below, observe the change in system response for the different initial conditions. For the third plot, observe the error inherent in the substitution. (For each plot, present your results with **time** in ***seconds*** and **angles** in ***degrees***). Use: $g = 9.81 \, {}^{m}\!/_{s^2}$, $l = 3m$ & $t = 0s \rightarrow 15s$.

  - Generate the first plot using the initial conditions: $\dot{\theta}(0) = 0$ and $\theta(0) = 0.05$, 0.1, 0.2, 0.5 & 1.0 rad. Overlay the *linearized* solution on the *exact* solution. Plot the response as: $\theta \quad vs. \quad time$.
  - Generate the second plot using the initial conditions: $\dot{\theta}(0) = 0$ and $\theta(0) = 1.0$, 2.0 & 3.0 rad. Overlay the *linearized* solution on the *exact* solution. Plot the response as: $\theta \quad vs. \quad time$.
  - Generate the third plot by plotting the relative error in the $\sin\theta \approx \theta$ assumption. Plot the relative error function as: $\dfrac{\theta - \sin\theta}{\theta} \quad vs. \quad \theta$ for $0 \le \theta \le 2\pi \; rad$

  Using the above plotted results, discuss the effects of the linearizing assumption on the predicted system response. Be sure to properly label and annotate your plots. (Don't forget UNITS!)

  Use linestyles and/or colors to differentiate the different cases. *You must include a copy of your <u>modified sdof function</u> in the results.*

The following is a MATLAB function you may use as a starting point. The example function solves the free response of an SDOF mck system. You will need to modify it to solve the exact and linearized pendulum problem given.

```
function prob_10F_example
% Solves an unforced translational MCK system.
%
% Important other related ODE functions: odeset

% Define time range.
t_start = 0;
t_stop  = 2;

t0 = [t_start ; t_stop];

% Define initial conditions.
y_disp = 0.5;
y_vel  = 0;

y0 = [y_vel ; y_disp];

% Assume default tolerance of 1.e-3;            % Accuracy
[t,y] = ode23(@sdof,t0,y0);

% plot(t,y(:,1)), title('SDOF time history (velocity)')

figure(1);
plot(t,y(:,2)), title('SDOF time history (displacement)')
xlabel('Time [sec]');
ylabel('Amplitude [m]');

end

%-----------------------------------------
%  Function to be integrated.
%
function ydot = sdof(t,y)

mass  =    10;  % kg
damp  =    10;  % N s / m
stiff = 16000;  % N / m

force =     0;  % N  -- no force being applied

xdot = y(1);
x    = y(2);

xdotdot = (force - damp .* xdot - stiff .* x) ./ mass;  % This is from the EOM.

ydot = [xdotdot ; xdot];

end
```