```
function varargout = gui_spring3mass3(varargin)
% GUI_SPRING3MASS3 MATLAB code for gui_spring3mass3.fig
% Simulates 3 masses connected by 3 springs to one wall
% How to use the GUI:
%    - Set desired parameters for 3 masses and 3 springs
%    - View oscillations corresponding to the three eigenvectors
%    - View oscillations corresponding to random initial conditions
%
% References:
%              www.youtube.com/watch?v=9NlGuQ26y80
%              www.youtube.com/watch?v=chh2XybEUjg
%              www.math.ust.hk/~machas/differential-equations.pdf

% Developed by Jeffrey R. Chasnov, Department of Mathematics,
%                         The Hong Kong University of Science &
 Technology.
% Uses the MATLAB code spring.m written by Gustavo Morales

% Last Modified by GUIDE v2.5 02-Jul-2013 17:26:09

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @gui_spring3mass3_OpeningFcn, ...
                   'gui_OutputFcn',  @gui_spring3mass3_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before gui_spring3mass3 is made visible.
function gui_spring3mass3_OpeningFcn(hObject, eventdata, handles,
 varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to gui_spring3mass3 (see VARARGIN)
global x1 x2 x3 v1 v2 v3
global m1 m2 m3 k1 k2 k3
global stop
stop=1;
```

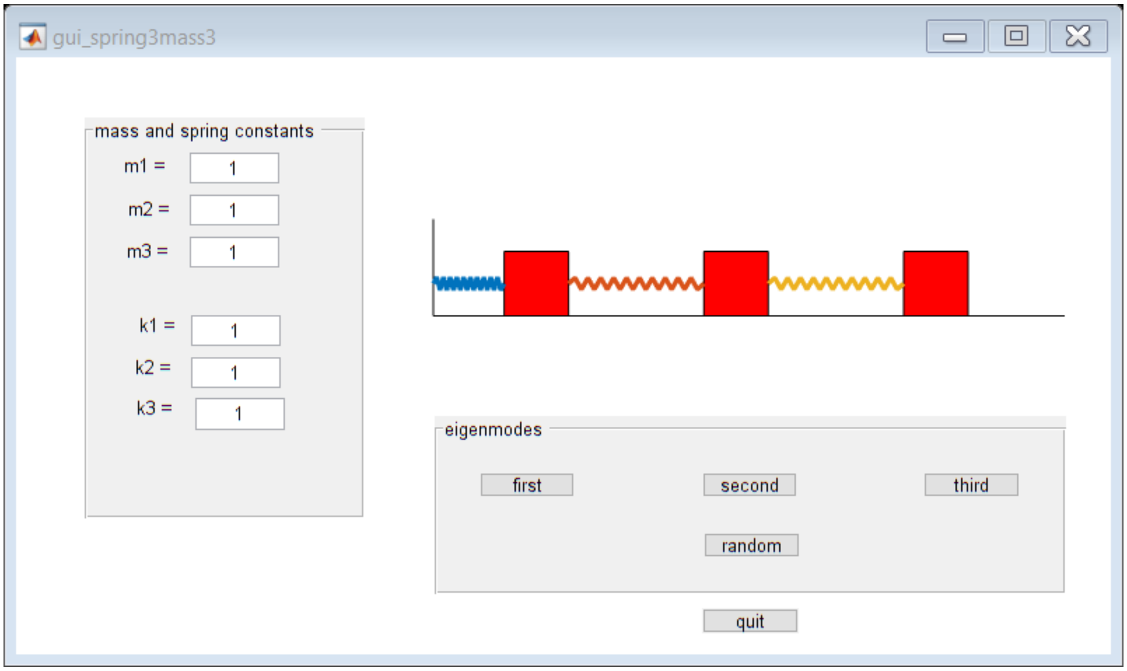
```
num_masses=3;
%default values
m1=1; m2=1; m3=1;
k1=1; k2=1; k3=1;
x1=0; x2=0; x3=0;
v1=0; v2=0; v3=0;
%length of springs
l1=1.1; l2=2.1; l3=2.1;
%set up initial figure of masses in equilibrium
axis([0 num_masses+l1+l2+l3+1.5 -1
 1]);axis equal;axis off;set(gcf,'Color','w')
line([0 0],[-0.5 1],'Color','k'); hold on;
%line([num_masses+l1+l2+l3 num_masses+l1+l2+l3],[-0.5 1],'Color','k');
line([0 num_masses+l1+l2+l3+1.5], [-0.5 -0.5],'Color','k');
%initial positions of masses
ypos=[-0.5 -0.5 0.5 0.5];
xpos=[l1+x1,1+l1+x1,1+l1+x1,l1+x1];
m1box=patch(xpos,ypos,'r');
xpos=[1+l1+l2+x2,2+l1+l2+x2,2+l1+l2+x2,1+l1+l2+x2];
m2box=patch(xpos,ypos,'r');
xpos=[2+l1+l2+l3+x3,3+l1+l2+l3+x3,3+l1+l2+l3+x3,2+l1+l2+l3+x3];
m3box=patch(xpos,ypos,'r');
%initial position of springs
ne = 10; a = 1; ro = 0.1;
[xs,ys]=spring(0,0,l1+x1,0,ne,a,ro);
 spring1=plot(xs,ys,'LineWidth',2);
[xs,ys]=spring(1+l1+x1,0,1+l1+l2+x2,0,ne,a,ro);
 spring2=plot(xs,ys,'LineWidth',2);
[xs,ys]=spring(2+l1+l2+x2,0,2+l1+l2+l3+x3,0,ne,a,ro);spring3=plot(xs,ys,'LineWidth
%eigenvectors
A=[ [-(k1+k2), k2, 0]/m1; [k2, -(k2+k3), k3]/m2; [0, k3, -k3]/m3 ];
[eigenvectors, eigenvalues]=eig(A);
[~,ix]=sort(diag(eigenvalues));
eigenvectors=eigenvectors(:,ix); %sort from higher frequency to lower
%data to share
handles.m1box=m1box;handles.m2box=m2box;handles.m3box=m3box;
handles.spring1=spring1;handles.spring2=spring2;handles.spring3=spring3;
handles.l1=l1;handles.l2=l2;handles.l3=l3;
handles.eigenvectors=eigenvectors;

% Choose default command line output for gui_spring3mass3
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
% UIWAIT makes gui_spring3mass3 wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = gui_spring3mass3_OutputFcn(hObject, eventdata,
 handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- runs oscillating spring
function run(handles)
global x1 x2 x3 v1 v2 v3
global m1 m2 m3 k1 k2 k3
global stop

N=64; T=(2*pi); dt=T/N; dt_pause=0.003;

%shared data
m1box=handles.m1box;m2box=handles.m2box;m3box=handles.m3box;
spring1=handles.spring1;spring2=handles.spring2;spring3=handles.spring3;
l1=handles.l1;l2=handles.l2;l3=handles.l3;

stop=0;
for i=1:intmax
    t_loopstart=tic();
    if stop
        break;
    end
    %fprintf('%g %f %f %f \n',i,x1,x2,x3);
    [t,y] = ode45(@(t,y) masses(t,y,m1,m2,m3,k1,k2,k3),...
                                               [0,dt],
[x1,v1,x2,v2,x3,v3]);
    x1=y(end,1); x2=y(end,3); x3=y(end,5);
    v1=y(end,2); v2=y(end,4); v3=y(end,6);
    xpos=[l1+x1,1+l1+x1,1+l1+x1,l1+x1];
    set(m1box,'xdata',xpos);
    xpos=[1+l1+l2+x2,2+l1+l2+x2,2+l1+l2+x2,1+l1+l2+x2];
    set(m2box,'xdata',xpos);
    xpos=[2+l1+l2+l3+x3,3+l1+l2+l3+x3,3+l1+l2+l3+x3,2+l1+l2+l3+x3];
    set(m3box,'xdata',xpos);
    [xs,ys]=spring(0,0,l1+x1,0); set(spring1,'xdata',xs,'ydata',ys);
    [xs,ys]=spring(1+l1+x1,0,1+l1+l2+x2,0);
 set(spring2,'xdata',xs,'ydata',ys);
    [xs,ys]=spring(2+l1+l2+x2,0,2+l1+l2+l3+x3,0);
 set(spring3,'xdata',xs,'ydata',ys);
    el_time=toc(t_loopstart);
    %disp(['Elapse time : ',num2str(el_time),' second']);
    pause(dt_pause-el_time);
end

function dy=masses(~,y,m1,m2,m3,k1,k2,k3,k4)
dy=zeros(6,1);
dy(1)=y(2);
dy(2)=(-(k1+k2)*y(1) + k2*y(3))/m1;
dy(3)=y(4);
dy(4)=(k2*y(1) - (k2+k3)*y(3) + k3*y(5))/m2;
dy(5)=y(6);
dy(6)=(k3*y(3)-k3*y(5))/m3;


function m1_Callback(hObject, eventdata, handles)
% hObject    handle to m1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of m1 as text
%        str2double(get(hObject,'String')) returns contents of m1 as a
 double
%if ~handles.stop
%StartStop_Callback(hObject, eventdata, handles);
%handles=guidata(handles.output);
global m1 m2 m3 k1 k2 k3
global stop
stop=1;
m1temp=str2double(get(hObject, 'String'));
if m1temp <= 0
    msgbox('Parameter out-of-range.  Choose m1 > 0.');
    set(hObject, 'String', num2str(m1));
    return;
else
    m1=m1temp;
end

A=[ [-(k1+k2), k2, 0]/m1; [k2, -(k2+k3), k3]/m2; [0, k3, -k3]/m3 ];
[eigenvectors, eigenvalues]=eig(A);
[~,ix]=sort(diag(eigenvalues));
eigenvectors=eigenvectors(:,ix); %sort from higher frequency to lower
handles.eigenvectors=eigenvectors;
guidata(hObject,handles)

% --- Executes during object creation, after setting all properties.
function m1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to m1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
 called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
 get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function m2_Callback(hObject, eventdata, handles)
% hObject    handle to m2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Hints: get(hObject,'String') returns contents of m2 as text
%        str2double(get(hObject,'String')) returns contents of m2 as a
 double
global m1 m2 m3 k1 k2 k3
global stop
stop=1;

m2temp=str2double(get(hObject, 'String'));
if m2temp <= 0
    msgbox('Parameter out-of-range.  Choose m2 > 0.');
    set(hObject, 'String', num2str(m2));
    return;
else
    m2=m2temp;
end

A=[ [-(k1+k2), k2, 0]/m1; [k2, -(k2+k3), k3]/m2; [0, k3, -k3]/m3 ];
[eigenvectors, eigenvalues]=eig(A);
[~,ix]=sort(diag(eigenvalues));
eigenvectors=eigenvectors(:,ix); %sort from higher frequency to lower
handles.eigenvectors=eigenvectors;
guidata(hObject,handles)


% --- Executes during object creation, after setting all properties.
function m2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to m2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
 called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
 get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function m3_Callback(hObject, eventdata, handles)
% hObject    handle to m3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of m3 as text
%        str2double(get(hObject,'String')) returns contents of m3 as a
 double
global m1 m2 m3 k1 k2 k3
global stop
stop=1;

m3temp=str2double(get(hObject, 'String'));
```

```
if m3temp <= 0
    msgbox('Parameter out-of-range.  Choose m3 > 0.');
    set(hObject, 'String', num2str(m3));
    return;
else
    m3=m3temp;
end

A=[ [-(k1+k2), k2, 0]/m1; [k2, -(k2+k3), k3]/m2; [0, k3, -k3]/m3 ];
[eigenvectors, eigenvalues]=eig(A);
[~,ix]=sort(diag(eigenvalues));
eigenvectors=eigenvectors(:,ix); %sort from higher frequency to lower
handles.eigenvectors=eigenvectors;
guidata(hObject,handles)

% --- Executes during object creation, after setting all properties.
function m3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to m3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
 called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
 get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function k1_Callback(hObject, eventdata, handles)
% hObject    handle to k1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of k1 as text
%        str2double(get(hObject,'String')) returns contents of k1 as a
 double
global m1 m2 m3 k1 k2 k3
global stop
stop=1;

k1temp=str2double(get(hObject, 'String'));
if k1temp <= 0
    msgbox('Parameter out-of-range.  Choose k1 > 0.');
    set(hObject, 'String', num2str(k1));
    return;
else
    k1=k1temp;
end

A=[ [-(k1+k2), k2, 0]/m1; [k2, -(k2+k3), k3]/m2; [0, k3, -k3]/m3 ];
[eigenvectors, eigenvalues]=eig(A);
```

```
[~,ix]=sort(diag(eigenvalues));
eigenvectors=eigenvectors(:,ix); %sort from higher frequency to lower
handles.eigenvectors=eigenvectors;
guidata(hObject,handles)

% --- Executes during object creation, after setting all properties.
function k1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to k1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
 called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
 get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function k2_Callback(hObject, eventdata, handles)
% hObject    handle to k2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of k2 as text
%        str2double(get(hObject,'String')) returns contents of k2 as a
 double
global m1 m2 m3 k1 k2 k3
global stop
stop=1;

k2temp=str2double(get(hObject, 'String'));
if k2temp <= 0
    msgbox('Parameter out-of-range.  Choose k2 > 0.');
    set(hObject, 'String', num2str(k2));
    return;
else
    k2=k2temp;
end

A=[ [-(k1+k2), k2, 0]/m1; [k2, -(k2+k3), k3]/m2; [0, k3, -k3]/m3 ];
[eigenvectors, eigenvalues]=eig(A);
[~,ix]=sort(diag(eigenvalues));
eigenvectors=eigenvectors(:,ix); %sort from higher frequency to lower
handles.eigenvectors=eigenvectors;
guidata(hObject,handles)

% --- Executes during object creation, after setting all properties.
function k2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to k2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
 called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
 get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function k3_Callback(hObject, eventdata, handles)
% hObject    handle to k3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of k3 as text
%        str2double(get(hObject,'String')) returns contents of k3 as a
 double
global m1 m2 m3 k1 k2 k3
global stop
stop=1;

k3temp=str2double(get(hObject, 'String'));
if k3temp <= 0
    msgbox('Parameter out-of-range.  Choose k3 > 0.');
    set(hObject, 'String', num2str(k3));
    return;
else
    k3=k3temp;
end

A=[ [-(k1+k2), k2, 0]/m1; [k2, -(k2+k3), k3]/m2; [0, k3, -k3]/m3 ];
[eigenvectors, eigenvalues]=eig(A);
[~,ix]=sort(diag(eigenvalues));
eigenvectors=eigenvectors(:,ix); %sort from higher frequency to lower
handles.eigenvectors=eigenvectors;
guidata(hObject,handles)

% --- Executes during object creation, after setting all properties.
function k3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to k3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns
 called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
 get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
% --- Executes on button press in first_eigenmode.
function first_eigenmode_Callback(hObject, eventdata, handles)
% hObject    handle to first_eigenmode (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global x1 x2 x3 v1 v2 v3
global stop

x1=handles.eigenvectors(1,1);
x2=handles.eigenvectors(2,1);
x3=handles.eigenvectors(3,1);
v1=0; v2=0; v3=0;

if stop; run(handles); end;

% --- Executes on button press in second_eigenmode.
function second_eigenmode_Callback(hObject, eventdata, handles)
% hObject    handle to second_eigenmode (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global x1 x2 x3 v1 v2 v3
global stop

x1=handles.eigenvectors(1,2);
x2=handles.eigenvectors(2,2);
x3=handles.eigenvectors(3,2);
v1=0; v2=0; v3=0;

if stop; run(handles); end;

% --- Executes on button press in third_eigenmode.
function third_eigenmode_Callback(hObject, eventdata, handles)
% hObject    handle to third_eigenmode (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global x1 x2 x3 v1 v2 v3
global stop

x1=handles.eigenvectors(1,3);
x2=handles.eigenvectors(2,3);
x3=handles.eigenvectors(3,3);
v1=0; v2=0; v3=0;

if stop; run(handles); end;

% --- Executes on button press in random.
function random_Callback(hObject, eventdata, handles)
% hObject    handle to random (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
%if running, stop it
global x1 x2 x3 v1 v2 v3
global stop
```

```
eigenvectors=handles.eigenvectors;
coefficients=2*(rand(3,1)-0.5);
xvec=eigenvectors*coefficients;
x1=xvec(1); x2=xvec(2); x3=xvec(3);
v1=0; v2=0; v3=0;

if stop; run(handles); end;

% --- Executes on button press in quit.
function quit_Callback(hObject, eventdata, handles)
% hObject    handle to quit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global stop
stop=1;
close all;
```



gui_spring3mass3

mass and spring constants

m1 = 1

m2 = 1

m3 = 1

k1 = 1

k2 = 1

k3 = 1

Published with MATLAB® R2017a