```
% C. Petersen, H. Werkle, Dynamik der Baukonstruktionen
% 2. Auflage, Springer Vieweg, Wiesbaden, 2018
%
% ML_09_1_Eigenfrequenzen_Staebe: Eigenwerte nicht-algebraischer
% Eigenwertprobleme - Stäbe analytisch
%
% Version 1.0, April 2018
% Softwareentwicklung:
% Andrei Firus, M.Eng (andrei.firus@gmail.com)

% Aufbau Eingabedateien: Eingabedatei nicht notwendig
%
% Ausgabedateien:
% Outputdatei_1: Eingaben- und Ergebnisübersicht

%---------------------------- EINGABEBLOCK --------------------------------
% Eingaben im Quellcode

l=9;                  % Stablänge [m]
mue=0.660;            % Masse pro Längeneinheit [t/m]
E=2.1*10^8;           % Elastizitätsmodul [kN/m^2]
I=9.375*10^-5;        % Trägheitsmoment [m^4]
n=10;                 % Anzahl der gesuchten Eigenfrequenzen

%!!!!!!!!!!!!!!!!!! DEFINITION DER AUFLAGERBEDINGUNGEN !!!!!!!!!!!!!!!!!!!
% Die Auflagerbedingungen sind durch die folgenden Angaben definiert:
% - Gelenkige Lagerung: "1"
% - Einspannung: "2"
% - freies Ende: "3"
Auflager_1=1;
Auflager_2=1;
%!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

% Darzustellende Eigenformen
EF_Plot_1=1;
EF_Plot_2=2;
EF_Plot_3=3;
%--------------------------------------------------------------------------

%------------------------- BERECHNUNGSBLOCK -----------------------------
% Lösungen der Frequenzgleichung
lambda=zeros(n,1);
if Auflager_1==1 && Auflager_2==1
    % Lagerungsfall gelenkig-gelenkig
    lambda(1)=pi;
    lambda(2)=2*pi;
    lambda(3)=3*pi;
    for i=4:1:n
        lambda(i)=i*pi;
    end
elseif (Auflager_1==1 && Auflager_2==2)||(Auflager_1==2 && Auflager_2==1)
    % Lagerungsfall gelenkig-eingespannt oder eingespannt-gelenkig
    lambda(1)=3.926602;
    lambda(2)=7.068582;
    lambda(3)=10.21018;
    for i=4:1:n
        lambda(i)=(i+1/4)*pi;
```

```
    end
elseif (Auflager_1==2 && Auflager_2==2)
    % Lagerungsfall eingespannt-eingespannt
    lambda(1)=4.730041;
    lambda(2)=7.853205;
    lambda(3)=10.99561;
    for i=4:1:n
        lambda(i)=(i+1/2)*pi;
    end
elseif (Auflager_1==2 && Auflager_2==3)||(Auflager_1==3 && Auflager_2==2)
    % Lagerungsfall eingespannt-frei oder frei-eingespannt
    lambda(1)=1.875104;
    lambda(2)=4.694091;
    lambda(3)=7.854760;
    for i=4:1:n
        lambda(i)=(i-1/2)*pi;
    end
end

% Eigenkreisfrequenzen, Eigenfrequenzen und Eigenschwingzeiten
omega=zeros(n,1);
Freq=zeros(n,1);
T=zeros(n,1);
for i=1:1:n
    omega(i)=(lambda(i)^2/l^2)*sqrt(E*I/mue);
    Freq(i)=omega(i)/(2*pi);
    T(i)=1/Freq(i);
end

% Eigenformen
dl=50*n+1;        % Schritt zur Ermittlung der Eigenform
x_plot=0:l/dl:l;
ef=zeros(n,length(x_plot));
x=zeros(1,length(x_plot));
if Auflager_1==1 && Auflager_2==1
    % Lagerungsfall gelenkig-gelenkig
for k=1:1:length(x)
        x(k)=0+(k-1)*1/dl*l;
end
for j=1:1:n
    for i=1:1:length(x)
        ef(j,i)=sin(lambda(j)*x(i)/l);
    end
end
elseif (Auflager_1==1 && Auflager_2==2)
    % Lagerungsfall gelenkig-eingespannt
    for k=1:1:length(x)
        x(k)=l-(k-1)*1/dl*l;
    end
for j=1:1:n
   for i=1:1:length(x)
        ef(j,i)=sin(lambda(j)*x(i)/l)-sinh(lambda(j)*x(i)/l)+...
            ((sinh(lambda(j))-sin(lambda(j)))/(cosh(lambda(j))-...
            cos(lambda(j))))*(cosh(lambda(j)*x(i)/l)-...
            cos(lambda(j)*x(i)/l));
    end
end
elseif (Auflager_1==2 && Auflager_2==1)
```

```
    % Lagerungsfall eingespannt-gelenkig
for k=1:1:length(x)
        x(k)=0+(k-1)*1/dl*l;
end
for j=1:1:n
     for i=1:1:length(x)
        ef(j,i)=sin(lambda(j)*x(i)/l)-sinh(lambda(j)*x(i)/l)+...
            ((sinh(lambda(j))-sin(lambda(j)))/(cosh(lambda(j))-...
            cos(lambda(j))))*(cosh(lambda(j)*x(i)/l)-...
            cos(lambda(j)*x(i)/l));
     end
 end
elseif (Auflager_1==2 && Auflager_2==2)
    % Lagerungsfall eingespannt-eingespannt
    for k=1:1:length(x)
        x(k)=0+(k-1)*1/dl*l;
    end
for j=1:1:n
        for i=1:1:length(x)
        ef(j,i)=sin(lambda(j)*x(i)/l)-sinh(lambda(j)*x(i)/l)+...
            ((sinh(lambda(j))-sin(lambda(j)))/(cosh(lambda(j))-...
            cos(lambda(j))))*(cosh(lambda(j)*x(i)/l)-...
            cos(lambda(j)*x(i)/l));
        end
end
elseif (Auflager_1==2 && Auflager_2==3)
    % Lagerungsfall eingespannt-frei
    for k=1:1:length(x)
        x(k)=0+(k-1)*1/dl*l;
    end
    for j=1:1:n
        for i=1:1:length(x)
        ef(j,i)=sin(lambda(j)*x(i)/l)-sinh(lambda(j)*x(i)/l)+...
            ((sinh(lambda(j))+sin(lambda(j)))/(cosh(lambda(j))+...
            cos(lambda(j))))*(cosh(lambda(j)*x(i)/l)-...
            cos(lambda(j)*x(i)/l));
        end
    end
elseif (Auflager_1==3 && Auflager_2==2)
    % Lagerungsfall frei-eingespannt
for k=1:1:length(x)
        x(k)=l-(k-1)*1/dl*l;
end
for j=1:1:n
        for i=1:1:length(x)
        ef(j,i)=sin(lambda(j)*x(i)/l)-sinh(lambda(j)*x(i)/l)+...
            ((sinh(lambda(j))+sin(lambda(j)))/(cosh(lambda(j))+...
            cos(lambda(j))))*(cosh(lambda(j)*x(i)/l)-...
            cos(lambda(j)*x(i)/l));
        end
end
end

% Normierung der Eigenformen auf das betragsgrößte Element
for j=1:1:n
    if max(ef(j,:))>abs(min(ef(j,:)))
       z1=max(ef(j,:));
       for i=1:1:length(ef)
```

```
          ef(j,i)=(ef(j,i)/z1);
        end
    else
        z1=min(ef(j,:));
        for i=1:1:length(ef)
          ef(j,i)=(ef(j,i)/z1);
        end
    end
end
%-------------------------------------------------------------------------

%------------------------ DARSTELLUNGSBLOCK ------------------------------
% Grafische Darstellung der Ergebnisse
name_fig1 = 'Ergebnisse';
fig1=figure('Name',name_fig1,'NumberTitle','off');
set(fig1,'Position',[200 300 800 500]);
plot(x_plot,ef(EF_Plot_1,:),x_plot,ef(EF_Plot_2,:),x_plot,ef(EF_Plot_3,:));
title1 = 'Eigenformen';
title(title1);
xlabel('Strukturkoordinate [m]');
ylabel('Eigenform [-]');
Legend_1 = [num2str(EF_Plot_1),'. Eigenform'];
Legend_2 = [num2str(EF_Plot_2),'. Eigenform'];
Legend_3 = [num2str(EF_Plot_3),'. Eigenform'];
legend(Legend_1,Legend_2,Legend_3);
grid on; zoom on;
%-------------------------------------------------------------------------

%---------------------------- AUSGABEBLOCK -------------------------------
% Ausgabe der Ergebnisse in eine Datei
fid = fopen('Outputdatei_1_Allgemein.txt', 'w');
fprintf(fid,...
    '%s\n','C. Petersen, H. Werkle, Dynamik der Baukonstruktionen');
fprintf(fid,...
    '%s\n','2. Auflage, Springer Vieweg, Wiesbaden, 2018');
fprintf(fid,...
    '%s\n','Softwareentwicklung: Andrei Firus (andrei.firus@gmail.com)');
fprintf(fid,'%s\n','Programm ML_09_1: Eingaben- und Ergebnisuebersicht');
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,...
    '%s\n','------------------------------------------------------------');
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,...
    '%s\n','EINGABEDATEN:');
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,'%s\n','Stablaenge [m]:');
fprintf(fid,'%d\n',l);
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,'%s\n','Masse pro Laengeneinheit [t/m]:');
fprintf(fid,'%d\n',mue);
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,'%s\n','Elastizitaetsmodul [kN/m^2]:');
fprintf(fid,'%d\n',E);
```

```
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,'%s\n','Traegheitsmoment [m^4]:');
fprintf(fid,'%d\n',I);
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,'%s\n','Anzahl der gesuchten Eigenformen [-]:');
fprintf(fid,'%d\n',n);
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,'%s\n','Lagerung:');
if Auflager_1==1
    A1="gelenkig";
elseif Auflager_1==2
    A1="eingespannt";
elseif Auflager_1==3
    A1="frei";
end
if Auflager_2==1
    A2="gelenkig";
elseif Auflager_2==2
    A2="eingespannt";
elseif Auflager_2==3
    A2="frei";
end
fprintf(fid, '%s\n', ' ');
fprintf(fid,'         %s |------------------------------| %s\n',A1,A2);
fprintf(fid, '%s\n', ' ');
fprintf(fid,...
    '%s\n','-------------------------------------------------------------');
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,...
    '%s\n','ERGEBNISSE:');
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,'%s\n','Berechnungsergebnisse:');
fprintf(fid,'%s \t %s \t  %s \t %s \t %s\n','Eigenform','lambda',...
    'Eigenkreisfrequenz [1/s]', 'Eigenfrequenz [Hz]',...
    'Eigenschwingzeit [s]');
for ii=1:1:n
    fprintf(fid,'%d \t %d \t %d \t %d \t %d\n',ii,lambda(ii),omega(ii),...
        Freq(ii),T(ii));
end
fprintf(fid, '%s\n', ' ');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf(fid,'%s\n','Eigenformen:');
Format= "%d \t ";
if n>1
    for ii=1:1:n-1
        Format=Format + '%d \t ';
    end
end
Format=Format + '%d\n';
for jj = 1:1:length(x_plot)
    fprintf(fid, Format, x_plot(jj),ef(:,jj)');
end
```

```
fclose(fid);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%--------------------------------------------------------------------------
```