

LAMPIRAN

A. Skrip Pemrograman Metode Identifikasi Kerusakan Struktur Berdasarkan Karakteristik Dinamik

A.1. Skrip Pemrograman Metode Identifikasi Kerusakan Untuk Sistem Rangka Batang (Contoh 3-Bentang, 4-Tingkat Rangka 2D)

```

function[resultat]=bay3etage4rangka(elel,deldam,zz)
% Program Perhitungan Sistem Struktur
% Rangka Batang 2D
%-----
% by
% Sendi aditya putra
% Young W. Kwoon
% Hyochoong Bang
% 2010
%-----
% ANALISA MODAL
% OUTPUT :
% 1. frek. natural dan Moda getar untuk struktur dengan dan tanpa kerusakan
% 2. Penentuan lokasi kerusakan
% 3. Penentuan Global Damage Index dengan perubahan frekuensi natural
% 4. Penentuan tingkat kerusakan
%-----
% Langkah-langkah :
%
%
% Deskripsi Variabel-Variabel :
% k = matriks kekakuan elemen
% kk = matriks kekakuan sistem
% m = matriks massa elemen
% mm = matriks massa sistem
% index = vektor hubungan DOFs dengan elemen yang ditinjau
% gcoord = matriks koordinat global untuk nodal tertentu
% disp = vektor perpindahan nodal
% elforce = vektor gaya elemen
% eldisp = perpindahan nodal elemen
% stress = vektor tegangan untuk tiap elemen
% elprop = matriks properti elemen
% nodes = matriks konektivitas modal untuk tiap elemen
% bcdof = vektor yang mengandung DOFs yang menjadi kondisi batas
% bcval = vektor yang mengandung nilai kondisi batas terhadap DOFs yang
% bersangkutan
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 1. ANALISA MODAL
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%
%-----
%-----
%

```

```

% U.N.D.A.M.A.G.E.D -- S.T.R.U.C.T.U.R.E
%
%-----
%-----
%-----
%-----
%-----
% INPUT DATA
%-----
%clc; clear all;
nel=40; % Jumlah total elemen
nnel=2; % Jumlah nodal per-elemen
ndof=2; % Jumlah DOF per-nodal
nnode=20; % Jumlah total nodal pada keseluruhan sistem struktur
sdof=nnode*ndof; % Total DOF pada keseluruhan sistem struktur
rho=7850; % Massa per unit volume (kg/m3)
elvh=(1:1:28); % List elemen vertikal dan horizontal
lvh=length(elvh);
eldiag=(29:1:40); % List elemen diagonal
ldiag=length(eldiag);

%-----
% KOORDINAT NODAL
%-----
gcoord(1,1)=0.0;gcoord(1,2)=0.0; % x,y - koordinat dari nodal ke-1
gcoord(2,1)=4.0;gcoord(2,2)=0.0; % x,y - koordinat dari nodal ke-2
gcoord(3,1)=8.0;gcoord(3,2)=0.0; % x,y - koordinat dari nodal ke-3
gcoord(4,1)=12.0;gcoord(4,2)=0.0; % x,y - koordinat dari nodal ke-4
gcoord(5,1)=0.0;gcoord(5,2)=3.0; % x,y - koordinat dari nodal ke-5
gcoord(6,1)=4.0;gcoord(6,2)=3.0; % x,y - koordinat dari nodal ke-6
gcoord(7,1)=8.0;gcoord(7,2)=3.0; % x,y - koordinat dari nodal ke-7
gcoord(8,1)=12.0;gcoord(8,2)=3.0; % x,y - koordinat dari nodal ke-8
gcoord(9,1)=0.0;gcoord(9,2)=6.0; % x,y - koordinat dari nodal ke-9
gcoord(10,1)=4.0;gcoord(10,2)=6.0; % x,y - koordinat dari nodal ke-10
gcoord(11,1)=8.0;gcoord(11,2)=6.0; % x,y - koordinat dari nodal ke-11
gcoord(12,1)=12.0;gcoord(12,2)=6.0; % x,y - koordinat dari nodal ke-12
gcoord(13,1)=0.0;gcoord(13,2)=9.0; % x,y - koordinat dari nodal ke-13
gcoord(14,1)=4.0;gcoord(14,2)=9.0; % x,y - koordinat dari nodal ke-14
gcoord(15,1)=8.0;gcoord(15,2)=9.0; % x,y - koordinat dari nodal ke-15
gcoord(16,1)=12.0;gcoord(16,2)=9.0; % x,y - koordinat dari nodal ke-16
gcoord(17,1)=0.0;gcoord(17,2)=12.0; % x,y - koordinat dari nodal ke-17
gcoord(18,1)=4.0;gcoord(18,2)=12.0; % x,y - koordinat dari nodal ke-18
gcoord(19,1)=8.0;gcoord(19,2)=12.0; % x,y - koordinat dari nodal ke-19
gcoord(20,1)=12.0;gcoord(20,2)=12.0; % x,y - koordinat dari nodal ke-20
%-----
% PROPERTI MATERIAL DAN GEOMETRIK PER ELEMEN
%-----
% Elemen Vertikal : W12x53
% Elemen Horizontal : W12x45
% Elemen Diagonal : W10x39
elprop(1:nel,1)=2e11; %Modulus elastisitas (N/m2) el. ke-1 sd. el.
ke-36
Aver=0.010064496; % Luasan penampang elemen vertikal (m2)
Ahor=8.516e-3; % Luasan penampang elemen horizontal (m2)
Adia=7.41934e-3; % Luasan penampang elemen diagonal (m2)
elver=[1 2 3 4 8 9 10 11 15 16 17 18 22 23 24 25]; % List el.2 vert.
(sesuai dg. penomoran)
elhor=[5 6 7 12 13 14 19 20 21 26 27 28]; % List el.2 hor. (sesuai dg.
penomoran)

```

```

eldia=[29:1:40]; % List el.2 dia. (sesuai dg. penomoran)
% Properti material untuk elemen vertikal
lver=length(elver);
for i=1:lver
    ii=elver(i);
    elprop(ii,2)=Aver;
end
%Properti material untuk elemen horizontal
lhor=length(elhor);
for j=1:lhor
    jj=elhor(j);
    elprop(jj,2)=Ahor;
end
%Properti material untuk elemen diagonal
ldia=length(eldia);
for k=1:ldia
    kk=eldia(k);
    elprop(kk,2)=Adia;
end
%-----
% KONEKTIFITAS NODAL
%-----
nodes(1,1)=1;nodes(1,2)=5; %nodal-nodal pada elemen ke-1
nodes(2,1)=2;nodes(2,2)=6; %nodal-nodal pada elemen ke-2
nodes(3,1)=3;nodes(3,2)=7; %nodal-nodal pada elemen ke-3
nodes(4,1)=4;nodes(4,2)=8; %nodal-nodal pada elemen ke-4
nodes(5,1)=5;nodes(5,2)=6; %nodal-nodal pada elemen ke-5
nodes(6,1)=6;nodes(6,2)=7; %nodal-nodal pada elemen ke-6
nodes(7,1)=7;nodes(7,2)=8; %nodal-nodal pada elemen ke-7
nodes(8,1)=5;nodes(8,2)=9; %nodal-nodal pada elemen ke-8
nodes(9,1)=6;nodes(9,2)=10; %nodal-nodal pada elemen ke-9
nodes(10,1)=7;nodes(10,2)=11; %nodal-nodal pada elemen ke-10
nodes(11,1)=8;nodes(11,2)=12; %nodal-nodal pada elemen ke-11
nodes(12,1)=9;nodes(12,2)=10; %nodal-nodal pada elemen ke-12
nodes(13,1)=10;nodes(13,2)=11; %nodal-nodal pada elemen ke-13
nodes(14,1)=11;nodes(14,2)=12; %nodal-nodal pada elemen ke-14
nodes(15,1)=9;nodes(15,2)=13; %nodal-nodal pada elemen ke-15
nodes(16,1)=10;nodes(16,2)=14; %nodal-nodal pada elemen ke-16
nodes(17,1)=11;nodes(17,2)=15; %nodal-nodal pada elemen ke-17
nodes(18,1)=12;nodes(18,2)=16; %nodal-nodal pada elemen ke-18
nodes(19,1)=13;nodes(19,2)=14; %nodal-nodal pada elemen ke-19
nodes(20,1)=14;nodes(20,2)=15; %nodal-nodal pada elemen ke-20
nodes(21,1)=15;nodes(21,2)=16; %nodal-nodal pada elemen ke-21
nodes(22,1)=13;nodes(22,2)=17; %nodal-nodal pada elemen ke-22
nodes(23,1)=14;nodes(23,2)=18; %nodal-nodal pada elemen ke-23
nodes(24,1)=15;nodes(24,2)=19; %nodal-nodal pada elemen ke-24
nodes(25,1)=16;nodes(25,2)=20; %nodal-nodal pada elemen ke-25
nodes(26,1)=17;nodes(26,2)=18; %nodal-nodal pada elemen ke-26
nodes(27,1)=18;nodes(27,2)=19; %nodal-nodal pada elemen ke-27
nodes(28,1)=19;nodes(28,2)=20; %nodal-nodal pada elemen ke-28
nodes(29,1)=1;nodes(29,2)=6; %nodal-nodal pada elemen ke-29
nodes(30,1)=5;nodes(30,2)=10; %nodal-nodal pada elemen ke-30
nodes(31,1)=9;nodes(31,2)=14; %nodal-nodal pada elemen ke-31
nodes(32,1)=13;nodes(32,2)=18; %nodal-nodal pada elemen ke-32
nodes(33,1)=2;nodes(33,2)=7; %nodal-nodal pada elemen ke-33
nodes(34,1)=6;nodes(34,2)=11; %nodal-nodal pada elemen ke-34
nodes(35,1)=10;nodes(35,2)=15; %nodal-nodal pada elemen ke-35
nodes(36,1)=14;nodes(36,2)=19; %nodal-nodal pada elemen ke-36
nodes(37,1)=3;nodes(37,2)=8; %nodal-nodal pada elemen ke-37

```

```

nodes(38,1)=7;nodes(38,2)=12; %nodal-nodal pada elemen ke-38
nodes(39,1)=11;nodes(39,2)=16; %nodal-nodal pada elemen ke-39
nodes(40,1)=15;nodes(40,2)=20; %nodal-nodal pada elemen ke-40
%-----
% KONDISI BATAS (PERLETAKAN)
%-----
bcdof=[1:1:8]; % dof ke-1 sd. 15 yang diberi kondisi batas
bcval=zeros(1,8); % nilai kondisi batas, perpindahan =0
lbc=length(bcdof);
%-----
% INISIALISASI - PEMBENTUKAN MATRIKS-MATRIKS NOL
%-----
ff=zeros(sdof,1); %Vektor gaya sistem
kk=zeros(sdof,sdof); %Matriks kekakuan sistem
mm=zeros(sdof,sdof); %Matriks massa sistem
index=zeros(nnel*ndof,1); %Vektor index
elforce=zeros(nnel*ndof,1); %Vektor gaya elemen
eldisp=zeros(nnel*ndof,1); %Vektor perpindahan nodal elemen
k=zeros(nnel*ndof,nnel*ndof); %Matriks kekakuan elemen
stress=zeros(nel,1); %Vektor tegangan untuk tiap elemen
%-----
% GAYA LUAR PADA NODAL TERTENTU
%-----
%ff(4)=-1000; % Nodal ke-2 diberi gaya sebesar 1000 kearah bawah
%-----
% LOOP UNTUK PENDEFINISIAN ELEMEN-ELEMEN
%-----
for iel=1:nel % Loop untuk keseluruhan elemen
    nd(1)=nodes(iel,1); %Nodal pertama yang terhubung dengan elemen ke-iel
    nd(2)=nodes(iel,2); %Nodal kedua yang terhubung dengan elemen ke-iel
    %
    x1=gcoord(nd(1),1);y1=gcoord(nd(1),2); %koordinat untuk nodal pertama
    %elemen ke-iel
    x2=gcoord(nd(2),1);y2=gcoord(nd(2),2); %koordinat untuk nodal kedua
    %elemen ke-iel
    %
    leng=sqrt((x2-x1)^2+(y2-y1)^2); %Panjang elemen ke-iel
    %
    % Perhitungan Sudut Antara Koordinata Lokal dan Global
    if (x2-x1)==0;
        beta=2*atan(1);
    else
        beta=atan((y2-y1)/(x2-x1));
    end
    %
    el=elprop(iel,1); %Ekstraksi modulus elastisitas untuk elemen ke-iel
    area=elprop(iel,2); %Ekstraksi luas penampang untuk elemen ke-iel
    %
    index=feeldof2D(nd,nnel,ndof); %Ekstraksi DOF yang berhubungan dengan
    %elemen ke-iel
    ipt=1; %ipt=1, for consistent mass matrix
        %ipt=2, for lumped mass matrix
    %iel=iel
    [k,m]=fetruss2(el,leng,area,rho,beta,ipt); %Perhitungan matriks
    kekakuan el.
    %
    kk=feasmbll(kk,k,index); %Menggabungkan semua matriks kekakuan elemen
    %menjadi matriks kekakuan sistem
    mm=feasmbll(mm,m,index); %Menggabungkan semua matriks kekakuan elemen

```

```

%menjadi matriks massa sistem
%
end
%-----
% KONDISI BATAS DAN PENYELESAIAN PERMASALAHAN
%-----
[kk,mm,ff]=feaplyc2c(kk,mm,ff,bcdof,bcval); % Pemberian kondisi batas
[kkk,mmm]=foreig(kk,mm,bcdof); % Penyederhanaan matriks sistem
%dengan mengeliminasi DOF yang berhubungan dengan kondisi batas
%
%disp=kk\ff; % Perhitungan solusi perpindahan nodal

%-----
% TAMPILKAN HASIL ANALISA FEM (ANALISA STATIK, MATRIKS KEKUAN DAN MASSA
% SISTEM)
%-----
%num=1:1:sdof;
%displ=[num' disp] % Tampilkan perpindahan tiap DOF
%
%kk
%mm
kkk;
mmm;
%-----
% HALIS ANALISA MODAL (MODA GETAR DAN NILAI EIGEN)
%-----
[V,D]=eig(kkk,mmm);
f=eig(kkk,mmm);
fsol=sqrt(f);
%
% Menampilkan hasil analisa modal
%
freqcy=[fsol]; % Menampilkan frekuensi natural per moda
V; %Menampilkan matriks moda getar
%
%-----
%
%-----
% D.A.M.A.G.E.D -- S.T.R.U.C.T.U.R.E
%
%-----
%
%-----
% INPUT DATA
%-----
%
% PROPERTI MATERIAL DAN GEOMETRIK PER ELEMEN (DAMAGED)
%-----
elpropd(1:nel,1)=2e11; %Modulus elastisitas (N/m2) el. ke-1 sd. el.
ke-36
%Aver=0.010064496; % Luasan penampang elemen vertikal (m2)
%Ahor=8.516e-3; % Luasan penampang elemen horizontal (m2)
%Adia=7.41934e-3; % Luasan penampang elemen diagonal (m2)
%elver=[1 2 4 5 7 8 10 11]; % List el.2 vert. (sesuai dg. penomoran)
%elhor=[3 6 9 12]; % List el.2 hor. (sesuai dg. penomoran)
%eldia=[13 14 15 16]; % List el.2 dia. (sesuai dg. penomoran)

```

```

% Properti material untuk elemen vertikal
lver=length(elver);
for i=1:lver
    ii=elver(i);
    elproprd(ii,2)=Aver;
end
%Properti material untuk elemen horizontal
lhor=length(elhor);
for j=1:lhor
    jj=elhor(j);
    elproprd(jj,2)=Ahor;
end
%Properti material untuk elemen diagonal
ldia=length(eldia);
for k=1:ldia
    kk=eldia(k);
    elproprd(kk,2)=Adia;
end
% Modifikasi untuk definisi sistem dengan kerusakan
%elel=16;%input('Elemen mana yang mengalami kerusakan=');
%deldam=input('Tingkat kerusakan pada elemen yang telah ditentukan(0-1)=');
temp=(1-deldam)*elproprd(elel,2);
elproprd(elel,2)=temp;
%-----
% INISIALISASI - PEMBENTUKAN MATRIKS-MATRIKS NOL
%-----
%s=length(bc dof); % Panjang vektor kondisi batas
ffd=zeros(sdof,1); %Vektor gaya sistem
kkd=zeros(sdof,sdof); %Matriks kekakuan sistem
mmd=zeros(sdof,sdof); %Matriks massa sistem
%kkk=zeros(sdof-s,sdof-s); %Matriks kekakuan sistem
%setelah eliminasi kondisi batas
%mmm=zeros(sdof-s,sdof-s); %Matriks massa sistem
%setelah eliminasi kondisi batas
indexd=zeros(nnel*ndof,1); %Vektor index
elforced=zeros(nnel*ndof,1); %Vektor gaya elemen
eldispd=zeros(nnel*ndof,1); %Vektor perpindahan nodal elemen
kd=zeros(nnel*ndof,nnel*ndof); %Matriks kekakuan elemen
stressd=zeros(nel,1); %Vektor tegangan untuk tiap elemen
%-----
% GAYA LUAR PADA NODAL TERTENTU
%-----
%ff(4)=-1000; % Nodal ke-2 diberi gaya sebesar 1000 kearah bawah
%-----
% LOOP UNTUK PENDEFINISIAN ELEMEN-ELEMEN
%-----
for iel=1:nel % Loop untuk keseluruhan elemen
    nd(1)=nodes(iel,1); %Nodal pertama yang terhubung dengan elemen ke-iel
    nd(2)=nodes(iel,2); %Nodal kedua yang terhubung dengan elemen ke-iel
    %
    x1=gcoord(nd(1),1);y1=gcoord(nd(1),2); %koordinat untuk nodal pertama
    %elemen ke-iel
    x2=gcoord(nd(2),1);y2=gcoord(nd(2),2); %koordinat untuk nodal kedua
    %elemen ke-iel
    %
    lengd=sqrt((x2-x1)^2+(y2-y1)^2); %Panjang elemen ke-iel
    %
    % Perhitungan Sudut Antara Koordinata Lokal dan Global
    if (x2-x1)==0;

```



```

% 2. PENENTUAN LOKASI KERUSAKAN
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% Lokasi kerusakan ditentukan dengan cara mendefinisikan vektor gaya residu
% Dengan mencocokkan DOF yang bersesuaian dengan komponen vektor gaya yang
% memiliki nilai  $\sim = 0$ , atau jauh lebih besar dibanding dengan komponen
% lainnya
%
%  $R_i = (-kkk + fdi * mmm) * Vid$ 
% fdi = nilai eigen moda ke-i keadaan rusak
% Vid = vektor moda getar ke-i keadaan rusak
% kkk, mmm = matriks kekakuan dan massa sistem keadaan tidak rusak
%-----
%
[RR]=damloc(kkk,mmm,fd,Vd);
%-----
% Menentukan DOF mana yang mengalami kerusakan
%-----
%disp('Cocokkan baris DOF pada vektor Df untuk menentukan DOF yang
mengalami');
%disp('kerusakan dengan vektor R. Apabila komponen R=0, DOF tersebut
tidak');
%disp('rusak dan sebaliknya');
RR=abs(RR);
Df;

%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 3. PENENTUAN GLOBAL DAMAGE INDEX DENGAN PERUBAHAN FREK. NATURAL
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% GDI ditentukan berdasarkan perubahan frekuensi natural. Dimana kontribusi
% tiap modal yang ada dimasukkan dalam perhitungan dengan menggunakan
% persentase massa modal efektif sebagai pengali tiap data modal yang
% digunakan.
%  $m_{eff}(n) = (Lh(n))^2 / mn$ 
% Data yang diperlukan : mmm (massa global sistem sebelum rusak) dan V
% (matriks moda getar)
% Cf. Dynamics of Structures by A.K. Chopra (modal mass effectif)
%
lh=mmm*V;
s=length(mmm);
Lh=zeros(1,s);
for p=1:s
    Lh=Lh+lh(p,:);
end
meff=zeros(1,s);
for q=1:s
    meff(q)=Lh(q)*Lh(q);
end
mtot=0;
for pp=1:s
    for qq=1:s
        mtot=mtot+mmm(pp,qq);
    end
end
end

```



```

pmeff=meff*(1/mtot);
% Perhitungan global damage index -->
di=zeros(1,s);
for ppp=1:s
    di(ppp)=pmeff(ppp)*(fsold(ppp)/fsol(ppp));
end
GD=0;
for qqq=1:s
    GD=GD+di(qqq);
end
%disp('Global Damage Index --->');
GDI=(1-GD)*100;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 4. PENENTUAN TINGKAT KERUSAKAN STRUKTUR
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
Vnew=zeros(s dof,s dof);
Vnew(lbc+1:s dof,lbc+1:s dof)=V; % Moda getar undamaged dalam matriks
%-----
%-----
% 4.1 SATU ELEMEN MENGALAMI KERUSAKAN
%-----
%-----
%
% Hal yang harus diperhatikan :
% Tentukan elemen mana yang rusak (identifikasi lokasi kerusakan)
%disp('Berikut adalah elemen yang mengalami kerusakan--->');
%ele1=14 % PENTING --> elemen mana yang mengalami kerusakan!
%
%-----
kg1=zeros(s dof,s dof); % matriks nol dengan ukuran s dof x s dof untuk kekakuan
elemen
mg1=zeros(s dof,s dof); % matriks nol dengan ukuran s dof x s dof untuk massa
elemen
nd1(1)=nodes(ele1,1);
nd1(2)=nodes(ele1,2);
x11=gcoord(nd1(1),1);y11=gcoord(nd1(1),2);
x12=gcoord(nd1(2),1);y12=gcoord(nd1(2),2);
leng1=sqrt((x12-x11)^2+(y12-y11)^2);
if (x12-x11)==0;
    betal=2*atan(1);
else
    betal=atan((y12-y11)/(x12-x11));
end
e11=elprop(ele1,1);
areal=elprop(ele1,2);
[k1,m1]=fetruss2(e11,leng1,areal,rho,betal,1);
[index1]=feeldof2D(nd1,nnel,ndof);
n1=length(index1);
for i=1:n1
    ii=index1(i);
    for j=1:n1;
        jj=index1(j);
        kg1(ii,jj)=k1(i,j);
    end
end

```

```

        mg1(ii,jj)=m1(i,j);
    end
end
kg1;
mg1;

% Lanjutkan proses perhitungan tingkat kerusakan
ndam=1; %jumlah elemen yang mengalami kerusakan
nneed=2*ndam; %jumlah data modal yang dibutuhkan untuk perhitungan
dof=[3:1:19]; %dof aktif (tanpa boundary condition)
Ai=zeros(nneed,1);
Bi=zeros(nneed,1);
delamb=zeros(nneed,1);
fu=fd-f; % Perubahan nilai eigen setelah terjadi kerusakan tiap moda
zz=input('pemilihan moda getar ke-zz+1 dalam proses identifikasi,zz=');
%-----
% untuk 1,2 2,3 3,4 4,5 5,6 (z)
for i=1:nneed
%   %mode=i+2;
%   %Vnew(:,i+2);
    Ai(i,1)=Vnew(:,i+lbc+zz)'*kg1*Vnew(:,i+lbc+zz);
    Bi(i,1)=f(i+zz)*Vnew(:,i+lbc+zz)'*mg1*Vnew(:,i+lbc+zz);
    delamb(i)=fu(i+zz);
end
%-----
% untuk 1,3 2,4 3,5 4,6 (z1)
%Ai(1,1)=Vnew(:,lbc+zz+1)'*kg1*Vnew(:,lbc+zz+1);
%Ai(2,1)=Vnew(:,lbc+zz+1+2)'*kg1*Vnew(:,lbc+zz+1+2);
%Bi(1,1)=f(zz+1)*Vnew(:,i+lbc+zz+1)'*mg1*Vnew(:,lbc+zz+1);
%Bi(2,1)=f(zz+1+2)*Vnew(:,i+lbc+zz+1+2)'*mg1*Vnew(:,lbc+zz+1+2);
%delamb(1)=fu(zz+1);
%delamb(2)=fu(zz+1+2);
%-----
% untuk 1,4 2,5 3,6 (z2)
%Ai(1,1)=Vnew(:,lbc+zz+1)'*kg1*Vnew(:,lbc+zz+1);
%Ai(2,1)=Vnew(:,lbc+zz+1+3)'*kg1*Vnew(:,lbc+zz+1+3);
%Bi(1,1)=f(zz+1)*Vnew(:,i+lbc+zz+1)'*mg1*Vnew(:,lbc+zz+1);
%Bi(2,1)=f(zz+1+3)*Vnew(:,i+lbc+zz+1+3)'*mg1*Vnew(:,lbc+zz+1+3);
%delamb(1)=fu(zz+1);
%delamb(2)=fu(zz+1+3);
%-----
% untuk 1,5 2,6 (z3)
%Ai(1,1)=Vnew(:,lbc+zz+1)'*kg1*Vnew(:,lbc+zz+1);
%Ai(2,1)=Vnew(:,lbc+zz+1+4)'*kg1*Vnew(:,lbc+zz+1+4);
%Bi(1,1)=f(zz+1)*Vnew(:,i+lbc+zz+1)'*mg1*Vnew(:,lbc+zz+1);
%Bi(2,1)=f(zz+1+4)*Vnew(:,i+lbc+zz+1+4)'*mg1*Vnew(:,lbc+zz+1+4);
%delamb(1)=fu(zz+1);
%delamb(2)=fu(zz+1+4);
%-----
% untuk 1,6 (z3)
%Ai(1,1)=Vnew(:,lbc+zz+1)'*kg1*Vnew(:,lbc+zz+1);
%Ai(2,1)=Vnew(:,lbc+zz+1+5)'*kg1*Vnew(:,lbc+zz+1+5);
%Bi(1,1)=f(zz+1)*Vnew(:,i+lbc+zz+1)'*mg1*Vnew(:,lbc+zz+1);
%Bi(2,1)=f(zz+1+5)*Vnew(:,i+lbc+zz+1+5)'*mg1*Vnew(:,lbc+zz+1+5);
%delamb(1)=fu(zz+1);
%delamb(2)=fu(zz+1+5);

L=[Ai Bi];

```

```

D1=((L'*L)^-1)*L'*delamb;
D11=L^-1*delamb;

%disp('Result');
%disp('-----');
%disp('1. Lokasi kerusakan');
%disp('-----');
%disp('Matriks yang berisi vektor gaya residual tiap modal');
RR;
%disp('Vektor yang menunjukkan korespondensi matriks RR dengan DOF')
Df;
bar(Df,RR(:,1));
set(gca,'color',[0.9 0.9 0.9]);
title('Identifikasi Lokasi Kerusakan Struktur','fontweight','bold');
xlabel('Degrees of Freedom','fontweight','bold')
ylabel('Residual Forces','fontweight','bold')
grid on;
%disp('2. Global Damage Index (dalam persentase)');
%disp('-----');
%disp('3. Identifikasi tingkat kerusakan (dalam persentase)');
%disp('-----');
damage=D1*100;
%disp(['delta-ki=' num2str(damage(1))]);
%disp(['delta-mi=' num2str(damage(2))]);
%f/1e5;
%fd/1e5;
%xx=(f-fd)/1e3;
%V;
resultat=[-damage(1);damage(2);GDI];

```

Berikut adalah fungsi-fungsi kecil yang mendukung skrip pemrograman diatas :

```

function [index]=feeldof2D(nd,nnel,ndof)
%Membentuk matriks index yang berhubungan dengan elemen yang berhubungan
%Yang mendefinisikan DOF untuk elemen tersebut
edof=nnel*ndof;
ii=nd(iel,1);
jj=nd(iel,2);
index(1)=2*ii-1;
index(2)=

function [k,m]=fetruss2(el,leng,area,rho,beta,ipt)
%-----
%Tujuan :
%Membentuk matriks kekakuan dan matriks massa untuk sistem Rangka 2D
%DOF (u1, v1, u2, v2)
%k - matriks kekakuan elemen 4x4
%m - matriks massa elemen 4x4
%el - modulus elastisitas
%area - luas penampang
%rho - densitas
%beta - sudut antara koordinat lokal dan global
%ipt = 1 - matriks massa konsisten
%ipt = 2 - matriks massa lumped
%-----

% MATRIKS KEKAKUAN
c=cos(beta);s=sin(beta);

```

```

k=(area*el/leng)*[c*c c*s -c*c -c*s;...
  c*s s*s -c*s -s*s;...
  -c*c -c*s c*c c*s;...
  -c*s -s*s c*s s*s]
%
% Matriks massa konsisten
if ipt==1
  m=(rho*area*leng/6)*[2*c*c 2*c*s c*c c*s;...
    2*cs 2 0 1;...
    1 0 2 0;...
    0 1 0 2]
  % Matriks Massa Lumped
else
  m=(rho*area*leng/2)*[1 0 0 0;...
    0 1 0 0;...
    0 0 1 0;...
    0 0 0 1]
end

function[kk]=feasmbll(kk,k,index)
%-----
%Tujuan :
%Membentuk Matriks Kekakuan Sistem Dari Matriks-Matriks Kekakuan Elemen
%kk - Matriks sistem
%k - Matriks elemen
%index - vektor yang mendefinisikan set DOF dengan elemen tertentu
%-----
edof=length(index);
for i=1:edof;
  ii=index(i);
  for j=1:edof
    jj=index(j);
    kk(ii,jj)=kk(ii,jj)+k(i,j);
  end
end

function[kk,mm,ff]=feaplyc2c(kk,mm,ff,bcdof,bcval)
%memberikan kondisi batas pada matriks sistem
n=length(bcdof);
sdof=size(kk);
for i=1:n
  c=bcdof(i);
  for j=1:sdof
    kk(c,j)=0; % Nullifying c-th row of the kk matrix
    kk(j,c)=0; % Nullifying c-th coloumn of the mm martrix
    mm(c,j)=0; % Nullifying c-th row of the kk matrix
    mm(j,c)=0; % Nullifying c-th coloumn of the mm martrix
  end
  kk(c,c)=1;
  mm(c,c)=1;
  ff(c)=bcval(i);
end

function [kkk,mmm,Df]=foreig(kk,mm,bcdof)
% menyederhanakan matriks kekakuan dan massa sistem dengan
% mengeliminasi DOF yang diberikan kondisi batas

```

```

% diperlukan dalam perhitungan modal analisis
a=bcdof';          % vektor kolom kondisi batas
sdof=length(kk);  % jumlah total DOF sistem
B=[1:1:sdof]';    % vektor urutan DOF sistem
n=length(B);
m=length(a);
A=zeros(n,1);
Df=zeros(sdof-m,1);
kkk=zeros(sdof-m,sdof-m);
mmm=zeros(sdof-m,sdof-m);
for i=1:m
    A(bcdof(i))=bcdof(i);
end
C=B-A;
C=sort(C,1);
for ii=1:(sdof-m)    % membentuk vektor yang hanya terdiri dari DOF
    Df(ii)=C(ii+m);  % setelah dilakukan eliminasi DOF kondisi batas
end
l=length(Df);
for p=1:l
    pp=Df(p);
    for q=1:l
        qq=Df(q);
        kkk(p,q)=kk(pp,qq);    % matriks kekakuan sistem setelah eliminasi
                                % DOF kondisi batas
        mmm(p,q)=mm(pp,qq);    % matriks massa sistem setelah eliminasi
                                % DOF kondisi batas
    end
end
end

```