## Laboratory 9

<table>
<thead>
<tr>
<th>Function</th>
<th>Variable Passed In</th>
<th>Size</th>
<th>Variable Passed Out</th>
<th>Size</th>
<th>Functions Called</th>
</tr>
</thead>
<tbody>
<tr>
<td>ud_elstf.m</td>
<td>L</td>
<td>1 by 1</td>
<td>KELE</td>
<td>12 by 12</td>
<td></td>
</tr>
<tr>
<td></td>
<td>A</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>lzz</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>lyy</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>J</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>v</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>truss</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ud_kltokg.m</td>
<td>L</td>
<td>1 by 1</td>
<td>KELEG</td>
<td>12 by 12</td>
<td>ud_trans.m</td>
</tr>
<tr>
<td></td>
<td>A</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td>ud_elstf.m</td>
</tr>
<tr>
<td></td>
<td>lzz</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>lyy</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>J</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>v</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>del</td>
<td>1 by 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>webdir</td>
<td>1 by 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>truss</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ud_kglobal.m</td>
<td>coord</td>
<td>nnodes by 3</td>
<td>KFF</td>
<td>nfdof by nfdof</td>
<td>ud_kltokg.m</td>
</tr>
<tr>
<td></td>
<td>nele</td>
<td>1 by 1</td>
<td>KSF</td>
<td>nsdof by nsdof</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ends</td>
<td>nele by 2</td>
<td>KFS</td>
<td>nsdof by nsdof</td>
<td></td>
</tr>
<tr>
<td></td>
<td>A</td>
<td>nele by 1</td>
<td>KSS</td>
<td>nsdof by nsdof</td>
<td></td>
</tr>
<tr>
<td></td>
<td>lzz</td>
<td>nele by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>lyy</td>
<td>nele by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>J</td>
<td>nele by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>nele by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>v</td>
<td>nele by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>truss</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>webdir</td>
<td>nele by 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>norder</td>
<td>nnodes by 6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>nfdof</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>nsdof</td>
<td>1 by 1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Bi-symmetrical framework element stiffness matrix.

\[
\begin{bmatrix}
F_{x1} \\
F_{y1} \\
M_{x1} \\
M_{y1} \\
F_{x2} \\
F_{y2} \\
M_{x2} \\
M_{y2}
\end{bmatrix} = \begin{bmatrix}
\frac{A}{L} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{12J}{L^3} & 0 & 0 & 0 & \frac{6J}{L^3} & 0 & \frac{12J}{L^3} \\
0 & \frac{6J}{L^3} & 0 & -\frac{6J}{L^3} & 0 & 0 & -\frac{12J}{L^3} & 0 \\
0 & \frac{12J}{L^3} & 0 & \frac{6J}{L^3} & 0 & 0 & 0 & -\frac{12J}{L^3} \\
0 & \frac{6J}{L^3} & 0 & \frac{4J}{L} & 0 & 0 & 0 & \frac{2J}{L} \\
0 & \frac{4J}{L} & 0 & -\frac{4J}{L} & 0 & 0 & 0 & \frac{2J}{L} \\
0 & \frac{2J}{L} & 0 & \frac{6J}{L^3} & 0 & 0 & 0 & 0 \\
0 & \frac{2J}{L} & 0 & \frac{6J}{L^3} & 0 & 0 & 0 & 0
\end{bmatrix} \begin{bmatrix}
\theta_{x1} \\
\theta_{y1} \\
\theta_{x2} \\
\theta_{y2} \\
\nu_{x1} \\
\nu_{y1} \\
\nu_{x2} \\
\nu_{y2}
\end{bmatrix}
\]

\[
\begin{bmatrix}
\mu_{1} \\
\mu_{2} \\
\mu_{3} \\
\mu_{4}
\end{bmatrix}
\]

\[E \quad \frac{A}{L} \quad \frac{12J}{L^3} \quad \frac{6J}{L^3} \quad \frac{12J}{L^3} \quad \frac{6J}{L^3} \quad \frac{4J}{L} \quad \frac{2J}{L} \quad \frac{2J}{L} \quad \frac{6J}{L^3} \quad \frac{4J}{L} \quad \frac{2J}{L} \quad \frac{2J}{L} \quad \frac{6J}{L^3} \quad \frac{4J}{L} \quad \frac{2J}{L} \quad \frac{2J}{L} \quad \frac{6J}{L^3}
\]

\[\text{Young's modulus} = E \quad \text{Shear modulus} = G\]
FUNCTION KELE = u_d_eidf(L,A,Izz,Iyy,J,E,v,truss)

FUNCTION KELEG = u_d_klog( L,A,Izz,Iyy,J,E,v,del,webdir, truss)

FUNCTIONS CALLED
< to be defined by the CEE361 student>

DICTIONARY OF VARIABLES

Input Information:
L = length of the element
A = cross sectional area of the element
Izz = moment of inertia about element's local z-z axis (strong axis)
Iyy = moment of inertia about element's local y-y axis (weak axis)
J = torsional constant
E = modulus of elasticity (Young's modulus)
v = Poisson's ratio

Local Information:
keleg(12,12) = element stiffness matrix in global coordinates
row,col = variables used for loop indices
ivar(i) = global DOF # corresponding to local DOF i of
the element being investigated
arow = variable used for array index

Output Information:
KFF(nfdof,nfdof) = [Kff], the free-free part of the global
stiffness matrix
KSF(nfdof,nfdof) = [Ksf], the support-free part of the global
stiffness matrix
KFS(nfdof,nfdof) = [Kfs], the free-support part of the global
stiffness matrix
KSS(nfdof,nfdof) = [Kss], the support-support part of the global
stiffness matrix

GOOD LUCK CEE 361 STUDENT!

node is "free" and norder will be negative
if the DOF of the node is "support".

nfdo = actual number of Free DOFs
nsdof = actual number of Support DOFs

Local Information:
ele = element # being investigated
L = length of element
dol3 = difference in element ele's x,y,z coordinates
dof = DOF # being investigated
keleg(12,12) = element stiffness matrix in global coordinates
row,col = variables used for loop indices
ivar(i) = global DOF # corresponding to local DOF i of
the element being investigated
arow = variable used for array index

Output Information:
KFF(nfdof,nfdof) = [Kff], the free-free part of the global
stiffness matrix
KSF(nfdof,nfdof) = [Ksf], the support-free part of the global
stiffness matrix
KFS(nfdof,nfdof) = [Kfs], the free-support part of the global
stiffness matrix
KSS(nfdof,nfdof) = [Kss], the support-support part of the global
stiffness matrix

GOOD LUCK CEE 361 STUDENT!
% GOOD LUCK CEE 361 STUDENT!

function [DEFL.REACT.ELE_FOR.AFLAG] = ud_3d1el(nnodes, coord, concen, fixity, nele, ends, A, Iz, J, Zz, Zyy, Ayy, Azz, E, v, Fy, webdir, beta_ang, w, truss, anatype);

% Function purpose:
% The function ud_3d1el performs a user defined three-dimensional
% first-order elastic analysis of a structural system.

% Functions Called
% < to be defined by the CEE361 student >

% Dictionary of Variables

% Input Information:

% nnodes = total number of nodes
% coord(i,1:3) = node i's coordinates
% coord(i,1) = X coordinate
% coord(i,2) = Y coordinate
% coord(i,3) = Z coordinate
% concen(i,1:6) = concentrated loads for node i's 6 d.o.f.
% concen(i,1) = force in global X direction
% concen(i,2) = force in global Y direction
% concen(i,3) = force in global Z direction
% concen(i,4) = moment about global X axis
% concen(i,5) = moment about global Y axis
% concen(i,6) = moment about global Z axis
% fixity(i,1:6) = prescribed displacements for node i's 6 d.o.f.
% Note: A free d.o.f. will have a value of NaN
% and hence, you will find the Matlab function
%iinan very useful.

% Examples:
% If fixity(15,3) is set to NaN, then node 15's
% Z-disp component is free;
% If fixity(2,6) is set to 0.0, then node 2's
% Z-rotation component is supported;
% If fixity(5,2) is set to -2.1, then node 5's
% Y-disp component is supported and defined
% with a settlement of -2.1 units.
% fixity(i,1) = prescribed disp. in global X direction
% fixity(i,2) = prescribed disp. in global Y direction
% fixity(i,3) = prescribed disp. in global Z direction
% fixity(i,4) = prescribed rotation about global X axis
% fixity(i,5) = prescribed rotation about global Y axis
% fixity(i,6) = prescribed rotation about global Z axis
% nele = total number of elements
% ends(1,1:4) = element i's nodal information
% ends(i,1) = start node
% ends(i,2) = finish node
% ends(i,3) = flag to indicate whether or not flexural
% moments are released at start node. ends(i,3)=0 not
% released (rigid connection); ends(i,3)=1 flexural
% moments are released (pinned connection)
% ends(i,4) = flag to indicate whether or not flexural
% moments are released at finish node. ends(i,4)=0 not
% released (rigid connection); ends(i,4)=1 flexural
% moments are released (pinned connection)
% A(i) = element i's cross sectional area
% Iz(i) = element i's moment of inertia about its local z-z axis
% Iyy(i) = element i's moment of inertia about its local y-y axis
% J(i) = element i's torsional constant
% Zz(i) = element i's plastic section modulus about local z-z axis
% Zyy(i) = element i's plastic section modulus about local y-y axis

% Ay(i) = element i's effective shear area along its local y-y axis
% Azz(i) = element i's effective shear area along its local z-z axis
% E(i) = element i's material elastic modulus. Young's Modulus
% v(i) = element i's material Poisson's ratio
% Fy(i) = element i's material yield strength
% webdir(i,1:3) = element i's unit web vector. This is a unit vector
% that defines the element's local y-y axis with respect
to the global coordinate system. It is based on the
structure's undeformed geometry.

% Note: MASTAN2 uses the following convention for
% defining a member's default web orientation:
% A vector defining the element's local y-axis
% will have a positive component in the global
% Y direction. If the element's local x-axis,
% its length axis, is aligned with the global Y
% axis, then element's local y-axis is aligned
% with global negative X axis. After this initial
% orientation, element i may be rotated about
% its local x-axis by the amount defined by
% its web rotation angle. beta_ang(i). The
% angle is in radians and assumes a right-hand
% convention about the local x-axis which is
% retained from the element's start node to its finish node.
% w(i,1:3) = element i's uniform load which references its
% local coordinate system
% w(i,1) = x component of uniform load
% w(i,2) = y component of uniform load
% w(i,3) = z component of uniform load
% truss = flag to indicate if structure is a truss or not
% truss = 0 System is not a truss
% truss = 1 System is a truss
% anatype = flag to indicate which type of analysis is requested
% anatype = 1 First-Order Elastic
% anatype = 2 Second-Order Elastic
% anatype = 3 First-Order Inelastic
% anatype = 4 Second-Order Inelastic
% anatype = 5 Elastic Buckling (Eigenvalue)
% anatype = 6 Inelastic Buckling (Eigenvalue)

% Local Information:
% < to be defined by the student >

% Output Information:
% DEFL(1,1:6) = node i's calculated 6 d.o.f. deflections
% DEFL(i,1) = displacement in X direction
% DEFL(i,2) = displacement in Y direction
% DEFL(i,3) = displacement in Z direction
% DEFL(i,4) = rotation about X direction
% DEFL(i,5) = rotation about Y direction
% DEFL(i,6) = rotation about Z direction
REACT(1,1:6) = reactions for supported node i's 6 d.o.f.
    REACT(1,1) = force in X direction
    REACT(1,2) = force in Y direction
    REACT(1,3) = force in Z direction
    REACT(1,4) = moment about X direction
    REACT(1,5) = moment about Y direction
    REACT(1,6) = moment about Z direction

ELR_FOR(1,1:12) = element i's internal forces and moments
    Note: All values reference the element's local coordinate system.
    ELR_FOR(1,1) = x-force at start node
    ELR_FOR(1,2) = y-force at start node
    ELR_FOR(1,3) = z-force at start node
    ELR_FOR(1,4) = x-moment at start node
    ELR_FOR(1,5) = y-moment at start node
    ELR_FOR(1,6) = z-moment at start node
    ELR_FOR(1,7) = x-force at end node
    ELR_FOR(1,8) = y-force at end node
    ELR_FOR(1,9) = z-force at end node
    ELR_FOR(1,10) = x-moment at end node
    ELR_FOR(1,11) = y-moment at end node
    ELR_FOR(1,12) = z-moment at end node

AFLAG = logical flag to indicate if a successful analysis has been completed
    AFLAG = 1  Successful
    AFLAG = 0  Unstable Structure
    AFLAG = inf  No analysis code available

Version 1.0/Student's Initials/Date of Modification

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Start by defining all output arrays to be empty
% NEFL = []; REACT = []; ELR_FOR = [];
% AFLAG = inf;
% STUDENT NOTE:
% In order for this routine to become fully active AFLAG must be changed.
% Student's code starts here...

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

GOOD LUCK CEE 361 STUDENT!