```
function [ SX,SY,SZ,SSTRAIN,X,Y,Z,Displacements,dimensions] = FEA_Routine( meshx_sample,mesh
%FEA_Routine: Performs mesh generation and solution of elasticity problem
%using 8-node hexahedral finite element analysis



%_________________________________________________________________
%| 8-Node Hexahedral Finite Element Solver
%|This function assembles a cuboidal mesh of 8-node hexahedral elements for
%input argument specified aspect ratios. It then imposes one of a predefined set
%of boundary conditions on the system and then solves for the displacement
%field. The system stiffness matrix is assembled  and the displacement
%field is solved for using
%Cholesky decomposition.
%
%Input arguments:
%
%meshx_sample:   The number of elements along the x -direction that the
%sample outside the stycast should have. Extra elements are added by the
%program for any length of crystal inside stycast.
%meshy: y -direction elements
%meshz: z-direction elements
%aspectab: sample length in x / sample length in y
%aspectac: sample length in x/ sample length in z
%bcmode: Boundary condition type. Takes values 1,2,3 corresponding to only
%edges of end faces clamped, ends embedded in stycast, end embedded but
%only bottom face constrained
%ex,ey,ez imposed strains along the x,y,z directions. Note that +ve
%corresponds to compression
%|material_type: takes values 1,2 correspondong to Sr$2$RuO$4$ and
%Isoptropic material
%styecast_ratio: Extra length of crystal at each end is embedded in
%styecast_ratio * Min(length in z, length in y)
%|
%| Output Variables:
%| SX, SY, SZ are mxnxp arrays specifying the X, Y, Z co-ordinates of the centre of the elem
% SSTRAIN is an mxnxpx6 arrays holding the strains 1-6 evaluated at the
% centre of each element in meshgrid format
%X,Y,Z specific co-ordinates of nodes in a format that will be described
%alongside their allocation
%Displacements holds the displacements in a format specified alongside allocation
%dimensions: Vector is the format [ total x-length, sample x-length, total
%elements in x, # el in y, #el in z]
%
%
%Required Functions:
```

```
%Preprocessor.m
%Boundary_Conditions.m
%Solver_v2.m
%Strain_calc.m
%
%
%
%
%
%-----------------------------------------------------------------

tic

%Calculate meshx_sty = # elements in x embedded in stycast, set equal to
%whichever of meshy, meshz is larger

if bcmode==2 || bcmode==3
meshx_sty=2*max(meshy,meshz);
else
    meshx_sty=0;
end

meshx=meshx_sample+2*meshx_sty; % Total number of elements in x
nel=meshx*meshy*meshz;    % Number of elements
nodex=meshx+1;            % Number of Nodes in x-direction
nodey=meshy+1;            % Number of Nodes in y-direction
nodez=meshz+1;            % Number of Nodes in z-direction
nodes=nodex*nodey*nodez;   % Number of Nodes
ndof=3*nodes;             % Number of degrees of freedom


% Allocate elasticity matrix E.

if material_type==1; % This corresponds to Sr$2$RuO$4$

        E11=2.32;
        E33=2.08;
        E23=0.71;
        E12=1.06;
        E44=0.657;
        E66=0.612;

        E=zeros(6,6);

        E(1,:)=[E11,E12,E23,0,0,0];
        E(2,:)=[E12,E11,E23,0,0,0];
```

```
        E(3,:)=[E23,E23,E33,0,0,0];
        E(4,:)=[0,0,0,E44,0,0];
        E(5,:)=[0,0,0,0,E44,0];
        E(6,:)=[0,0,0,0,0,E66];

        E=E.*1e11;

elseif material_type==2 % Isotropic material

        v=0.3;
        Emag=1e11;

      E=Emag/((1+v)*(1-2*v))*[1-v ,  v , v ,  0      ,    0    ,    0     ;
                              v  ,1-v , v ,  0      ,    0    ,    0     ;
                              v  ,  v ,1-v,  0      ,    0    ,    0     ;
                              0  ,  0 , 0 ,(1-2*v)/2,    0    ,    0     ;
                              0  ,  0 , 0 ,  0      ,(1-2*v)/2,    0     ;
                              0  ,  0 , 0 ,  0      ,    0    ,(1-2*v)/2];

else

    error('Unrecognised value for material type');

end

%____PREPROCESSOR_________________________________________________________%
%Generates mesh co-ordinates and topology. See Preprocessor.m for full
%desciption of method and output variables

[Cor , Pos, sample,total]=Preprocessor( aspectab,aspectac,meshx_sample,...
 meshx_sty,meshx,meshy,meshz,bcmode,styecastratio,nodes,nodex,nodey,nodez);


%_________________________________________________________________________%


%__Generate Boundary Conditions___________________________________________
%See Boundary_Conditions.m for full description
[Re,R,dofConstraints,P]=Boundary_Conditions(Pos,Cor,ex, ey, ez,nodes,...
    nel,ndof,bcmode,sample,total);


%_________________________________________________________________________

%___Solver_______________________________________________________________%
%Generates system stiffness matrix, enforces boundary conditions and solves
```

```
%for nodal displacements. See Solver_v2.m for details.

[D,sparseKsis]=Solver_v2(Cor,Pos,R,E,dofConstraints,P,nel,ndof);

%______________________________________________________________________%


%___Strain Calculation_________________________________________________%
%Calculates strains from displacement field.

 [strain, Displacements]=Strain_calc(Cor,Pos,R,nel,D);

 %______________________________________________________________________%



 %Generates arrays X,Y,Z that specify node co-ordinates in the form
 % x coordinate of node in element m with local node number n = X(n,m)


  for s=1:nel
X(:,s)=Cor(Pos(s,:)',1);
end

for s=1:nel
Y(:,s)=Cor(Pos(s,:)',2);
end

for s=1:nel
Z(:,s)=Cor(Pos(s,:)',3);
end


% Calculate the centre of each element: x coordinate at centre of element
% m=xcen(m)

xcen=(mean(X));
ycen=mean(Y);
zcen=mean(Z);

%Element centre coordinates and strains are put into the arrays SX, SY, SZ
%described in the header. These are of the form required by the matlab
%slice function for volume exploration

dimensions1=[total(1),total(2),total(3),meshx,meshy,meshz,nel];
```

```
[SX,SY,SZ,SSTRAIN]=Sliceform(dimensions1,strain,xcen,ycen,zcen);


%Allocate vector holding simulation dimensions in form
% [ sample length in x, sl y, slz, total length in x,tl y, tlz, meshx...
%,meshy, meshyz, nel]
dimensions=[sample,total,meshx,meshy,meshz,nel];



toc



end
```

```
Error using FEA_Routine (line 63)
Not enough input arguments.
```