```
function [TriFEM] = HW3_1TriFEM

% compute potential and C/epsilon0 for 2D region modeled with tri
 cells
%
% July 28, 2018   A. F. Peterson
% Modified September 14, 2018 by Caitlyn Caggia

% read fem mesh from file 'cylfil.txt'
%
% mesh should be organized so that interior nodes appear first,
 followed
% by nodes on the outer boundary (the driven boundary), followed by
 nodes
% on the inner boundary

nnodes = dlmread('cylfil.txt','', [0,0,0,0]);
ncells = dlmread('cylfil.txt','', [0,1,0,1]);
ninner = dlmread('cylfil.txt','', [0,2,0,2]);
nouter = dlmread('cylfil.txt','', [0,3,0,3]);

xy=dlmread('cylfil.txt','', [1,1,nnodes,2]);

nstart=nnodes + 1;
nend=nstart + ncells - 1;
pcetond=dlmread('cylfil.txt','', [nstart,1,nend,3]);

nstart=nend + 1;
nend=nstart + ncells - 1;
er=dlmread('cylfil.txt','', [nstart,1,nend,1]);

% initialize variables

nunks = nnodes - ninner - nouter;

Wtilda=zeros(nnodes);
W=zeros(nunks);
V=zeros(nunks,1);
elem(3,3)=0;

% loop through the cells, filling global matrix one cell at a time

for icell=1:ncells

    n1=pcetond(icell,1);
    n2=pcetond(icell,2);
    n3=pcetond(icell,3);

    %    compute 3 by 3 element matrix

    x(1)=xy(n1,1);  y(1)=xy(n1,2);
    x(2)=xy(n2,1);  y(2)=xy(n2,2);
```

```
    x(3)=xy(n3,1);  y(3)=xy(n3,2);

    b(1)=y(2)-y(3);
    b(2)=y(3)-y(1);
    b(3)=y(1)-y(2);

    c(1)=x(3)-x(2);
    c(2)=x(1)-x(3);
    c(3)=x(2)-x(1);

    Area = abs(b(3)*c(1) - b(1)*c(3))*0.5;

    for ii=1:3
        for jj=1:3
            elem(ii,jj)=(b(ii)*b(jj)+c(ii)*c(jj))*er(icell)/Area/4;
        end
    end

    %    add contributions from cell 'icell' to global matrices

    for ii=1:3
        ig=pcetond(icell,ii); % 'ig' is the global node for 'ii'
        for jj=1:3
            jg=pcetond(icell,jj);  % 'jg' is the global node for 'jj'
            Wtilda(ig,jg) = Wtilda(ig,jg) + elem(ii,jj);
            if(ig <= nunks) % test function at interior node
                if(jg <= nunks) % basis function at interior node
                    W(ig,jg) = W(ig,jg) + elem(ii,jj);
                elseif(jg <= nunks+nouter) % basis function on outer
 bnd
                    V(ig) = V(ig) - elem(ii,jj);
                end
            end
        end
    end
end

%  solve the system of equations to find the potential function

Pot = W\V;


%  compute the capacitance

Pinner = zeros(ninner,1);
Pouter = ones(nouter,1);
phi = [Pot; Pouter; Pinner];
V0 = 1;
Cap = 1/V0^2 * phi' * Wtilda * phi;


%  write results to file 'potfil.txt'

% str = ['Capacitance/epsilon0 pul = ',num2str(Cap)];  disp(str);
```

```
fid = fopen('potfil.txt', 'wt');

%   fprintf(fid,'%s\n\n',str);
str = 'node    Potential';
fprintf(fid,'%s\n\n',str);
for ii=1:nunks
    fprintf(fid,'%6d %15.14g\n',ii, Pot(ii));
end
fclose(fid);

TriFEM = Cap;
end
```

Published with MATLAB® R2018a