```
%=========================================================================
%  Function [Tint] = Tint(u,v)
%
%  integrand for S element matrix integral: grad dot grad
%=========================================================================

function [Tint] = Tint(u,v)

      global itest ibasis

      [tu, tv, t] = basis(itest,u,v);
      [bu, bv, b] = basis(ibasis,u,v);
      [rdet, rjac] = rjacob(u,v);

   Tint = t*b*rdet;

end

%=========================================================================
%  Function [b, bu, bv] = basis(ibas,u,v)
%
%  Basis function & first derivatives at (u,v) in standard triangle
%=========================================================================

function [bu, bv, b] = basis(ibas,u,v)

%     quadratic Lagrangian bases on a standard triangle
%
%     (u,v,w) are the simplex variables

      w=1-u-v;
      if(ibas == 1)
         b=(2*u-1)*u;
         bu=4*u-1;
         bv=0;
      elseif(ibas == 2)
         b=(2*v-1)*v;
         bu=0;
         bv=4*v-1;
      elseif(ibas == 3)
         b=(2*w-1)*w;
         bu=-3+4*u+4*v;
         bv=-3+4*u+4*v;
      elseif(ibas == 4)
         b=4*v*w;
         bu=-4*v;
         bv=4-4*u-8*v;
      elseif(ibas == 5)
         b=4*u*w;
         bu=4-8*u-4*v;
         bv=-4*u;
      elseif(ibas == 6)
```

```
         b=4*u*v;
         bu=4*v;
         bv=4*u;
      else
         disp('ibas out of range in BASIS: '); disp(ibas)
      end
end

%=======================================================================
%   rjacob: jacobian matrix for curved triangle patch transformation
%=======================================================================

function [rdet,rjac] = rjacob(u,v)

%     Jacobian matrix associated with the mapping from a curved
%     patch in 2D space to a reference cell of triangular shape.
%
%   variables defined on input:
%     Local coordinates are (u,v)
%     Cell index is 'ncell'
%
%   variables produced as output:
%     determinant is 'rdet'
%     Jacobian matrix is 'rjac'

      global pcetond xy;
      global icell;

      x(6)=0; y(6)=0;

      for i=1:6
     j=pcetond(icell,i);
     x(i)=xy(j,1);
     y(i)=xy(j,2);
      end

      rjac(1,1) = 4.*x(5)-3.*x(3)-x(1) + 4.*u*(x(1)+x(3)-2.*x(5)) +
  ...
                  4.*v*(x(3)+x(6)-x(5)-x(4));

      rjac(1,2) = 4.*y(5)-3.*y(3)-y(1) + 4.*u*(y(1)+y(3)-2.*y(5)) +
  ...
                  4.*v*(y(3)+y(6)-y(5)-y(4));

      rjac(2,1) = 4.*x(4)-3.*x(3)-x(2) + 4.*u*(x(3)+x(6)-x(5)-x(4)) +
  ...
                  4.*v*(x(2)+x(3)-2.*x(4));

      rjac(2,2) = 4.*y(4)-3.*y(3)-y(2) + 4.*u*(y(3)+y(6)-y(5)-y(4)) +
  ...
                  4.*v*(y(2)+y(3)-2.*y(4));

%     calculate determinant
```

```
        rdet=rjac(1,1)*rjac(2,2) - rjac(1,2)*rjac(2,1);
end

%=========================================================================
%=========================================================================
```

Published with MATLAB® R2018a