```
function [Q,Z,E,A,C,Ahat,s,t,k] = Embed(E,A,C,Q,Z,tol)
%
%  function [Q,Z,E,A,df,t] = Embed(E,A,Q,Z,tol)
%  embeds a full row rank pencil  [A22 A23]-s[E22 E23]
%  to a form Q'([A22 A23]-s[E22 E23])Z
%                [       Ahat       ]
%  that is unimodular, by making use of the staircase algorithm
%  and a constant matrix Ahat. The rank check tolerance is tol.
%  The routine returns the unitary transformations Q and Z
%  the sets of dimensions s, t and k of the diagonal blocks
%  and the constant embedding Ahat
%
mn=size(E);m=mn(1);n=mn(2);mcur=1;ncur=1;
% The block dimensions of the stairs in A are t by s
s=zeros(1,0);t=zeros(1,0);
% The block sizes of the embedding Ahat are by k by s
Ahat=zeros(0,0);k=zeros(0,1);
% Put the m x n full row rank pencil A-sE in staircase form
while mcur <= m
   % First compress the columns of the trailing block E
   [Qright,Er,rE]=ColCompR(E(mcur:m,ncur:n),tol);
   E(mcur:m,ncur:n)=Er;
   E(1:mcur-1,ncur:n)=E(1:mcur-1,ncur:n)*Qright;
   A(1:m,ncur:n)=A(1:m,ncur:n)*Qright;
   C(:,ncur:n)=C(:,ncur:n)*Qright;
   Z(:,ncur:n)=Z(:,ncur:n)*Qright;
   % Then compress the columns of the new leading block A
   [Qleft,Ar,rA]=RowCompT(A(mcur:m,ncur:n-rE),tol);
   A(mcur:m,ncur:n-rE)=Ar;
   A(mcur:m,n-rE+1:n)=Qleft'*A(mcur:m,n-rE+1:n);
   E(mcur:m,n-rE+1:n)=Qleft'*E(mcur:m,n-rE+1:n);
   Q(:,mcur:m)=Q(:,mcur:m)*Qleft;
   % Complete the rows to a unimodular pencil
   snew=n-ncur+1-rE;tnew=rA;knew=snew-tnew;
   Ahat(sum(k)+1:sum(k)+knew,sum(s)+1:sum(s)+snew)=RowOrthCompl(Ar(1:rA,:));
   % Now update the dimensions
   s=[s,snew];t=[t,tnew];k=[k,knew];
   mcur=mcur+rA;ncur=n-rE+1;
end
snew=n-ncur+1;tnew=0;knew=snew-tnew;
% The last embedding block is always an identity matrix of size snew
Ahat(sum(k)+1:sum(k)+knew,sum(s)+1:sum(s)+snew)=eye(snew,snew);
s=[s,snew];t=[t,tnew];k=[k,knew];
```