```
% To implement Newton's algorithm.
% [xs,fs,k] = newton('f_rosen','g_rosen','h_rosen',[0; 2],0.1,1e-6);
function [xs,fs,k] = newton(fname,gname,hname,x0,dt,epsi)
format compact
format long
k = 1;
xk = x0;
gk = feval(gname,xk);
Hk = feval(hname,xk);
[V,D] = eig(Hk);
di = diag(D);
dmin = min(di);
if dmin > 0
   Hki = V*diag(1./di)*V';
else
   bt = dt - dmin;
   Hki = V*diag((1+bt)./(di+bt))*V';
end
dk = -Hki*gk;
ak = bt_lsearch2018(xk,dk,fname,gname);
adk = ak*dk;
er = norm(adk);
xs = [xk];
while er >= epsi
  xk = xk + adk;
  gk = feval(gname,xk);
  Hk = feval(hname,xk);
  [V,D] = eig(Hk);
  di = diag(D);
  dmin = min(di);
  if dmin > 0
     Hki = V*diag(1./di)*V';
  else
     bt = dt - dmin;
     Hki = V*diag((1+bt)./(di+bt))*V';
  end
  dk = -Hki*gk;
  ak = bt_lsearch2018(xk,dk,fname,gname);
  adk = ak*dk;
  er = norm(adk);
  k = k + 1;
  xs = [xs xk];
end
disp('solution:')
xs = [xs  (xk + adk)];
xs(:,end)
disp('objective function at solution point:')
fs = feval(fname,xs(:,end))
format short
disp('number of iterations performed:')
k
```

Published with MATLAB® R2018b