```
%**************************************************************************
%*     Example 14.3                                                       *
%*     filename: ch14pr03.m                                               *
%*     program listing number: 14.3                                       *
%*                                                                        *
%*     This program solves 2-dimensional Laplace equation using the SOR   *
%*     method.                                                            *
%*                                                                        *
%*     Programed by Ryoichi Kawai for Computational Physics Course.       *
%*     Last modification:  04/16/2017.                                    *
%**************************************************************************
close all;
clear all;

% parameters
a=1.0;
b=1.0;
V=1.0;

% spacial domain
Nx=201; % number of grids
Ny=101;
dx=0.1; % spacial step
dy=0.1;
x=linspace(-b,b,Nx);
y=linspace(0,a,Ny);

% time step
h=1./4.;

% SOR parameter
r=0.5*(cos(pi/Nx)+cos(pi/Ny));
w=2./(1.+sqrt(1-r^2));

%tolerence
tol=1.e-9;

% sampling interval
M=10;

% initial profile
phi0=zeros(Ny,Nx);
phi0(:,1)=V;
phi0(:,Nx)=V;
phi0(1,:)=0;
phi0(Ny,:)=0;

% allocate arrays
phi1=phi0;

k=0;
diff=realmax();
figure(1)
while diff>tol
    k=k+1;
    for i=2:Ny-1
        for j=2:Nx-1
            phi1(i,j)=(1.0-w)*phi0(i,j)+w*h*(phi1(i-1,j)+phi0(i+1,j)+phi1(i,j-1)+phi0(i,j+1));
        end
    end
    if mod(k,M)==0  % record the results
        s=(phi1-phi0).^2;
        diff=sum(s(:));
        fprintf('%d : diff=%14.6e\n',k,diff);
        pcolor(phi1); axis equal tight; shading interp;
        xlabel('x');
        ylabel('y');
        drawnow;
    end
    phi0=phi1;
end
colorbar

% Plot time evolution as 3D plot
figure(2)
contour(x,y,phi1);
hold on;
[X,Y]=meshgrid(x(6:10:Nx-1),y(6:10:Ny-1));
[GX,GY]=gradient(phi1);
G=sqrt(GX(6:10:Ny-1,6:10:Nx-1).^2+GY(6:10:Ny-1,6:10:Nx-1).^2);
VX=GX(6:10:Ny-1,6:10:Nx-1)./G;VY=GY(6:10:Ny-1,6:10:Nx-1)./G;
quiver(X,Y,VX,VY,0.5)
hold off
axis equal tight;
xlabel('x');
ylabel('y');
```