Matlab Program for Second Order FD Solution to Poisson's Equation

Code:

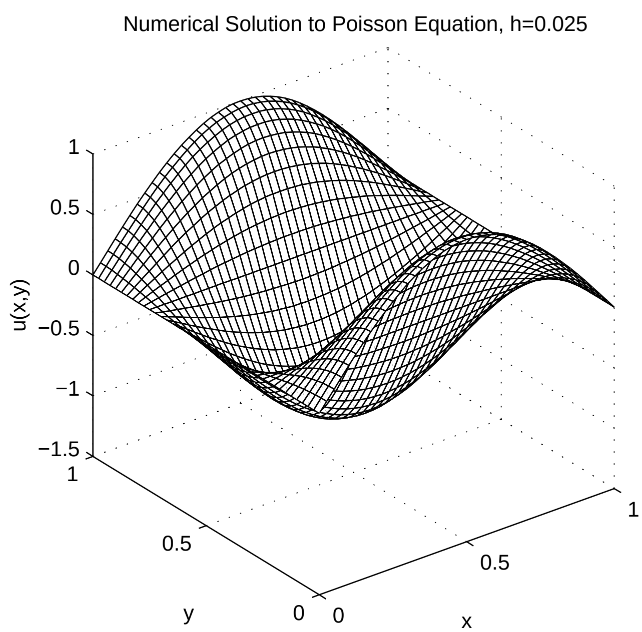
```
% Numerical approximation to Poisson's equation over the square [a,b]x[a,b] with
% Dirichlet boundary conditions.  Uses a uniform mesh with (n+2)x(n+2) total
% points (i.e, n x n interior grid points).
% Input:
%        pfunc : the RHS of poisson equation (i.e. the Laplacian of u).
%        bfunc : the boundary function representing the Dirichlet B.C.
%          a,b : the interval defining the square
%            n : n+2 is the number of points in either direction of the mesh.
% Ouput:
%            u : the numerical solution of Poisson equation at the mesh points.
%          x,y : the uniform mesh.
%
function [u,x,y] = fd2poisson(pfunc,bfunc,a,b,n)

h = (b-a)/(n+1);   % Mesh spacing

[x,y] = meshgrid(a:h:b);   % Uniform mesh, including boundary points.

% Compute u on the boundary from the Dirichlet boundary condition
ub = zeros(n,n);
idx = 2:n+1;
idy = 2:n+1;
% West and East boundaries need special attention
ub(:,1) = feval(bfunc,x(idx,1),y(idy,1));            % West Boundary
ub(:,n) = feval(bfunc,x(idx,n+2),y(idy,n+2));        % East Boundary
% Now the North and South boundaries
ub(1,1:n) = ub(1,1:n) + feval(bfunc,x(1,idx),y(1,idy));
ub(n,1:n) = ub(n,1:n) + feval(bfunc,x(n+2,idx),y(n+2,idy));
% Convert ub to a vector using column reordering
ub = (1/h^2)*reshape(ub,n*n,1);

% Evaluate the RHS of Poisson's equation at the interior points.
f = feval(pfunc,x(idx,idy),y(idx,idy));
% Convert f to a vector using column reordering
f = reshape(f,n*n,1);

% Create the D2x and D2y matrices
z = [-2;1;zeros(n-2,1)];
D2x = 1/h^2*kron(toeplitz(z,z),eye(n));
D2y = 1/h^2*kron(eye(n),toeplitz(z,z));

% Solve the system
u = (D2x + D2y)\(f-ub);
% Convert u from a column vector to a matrix to make it easier to work with
% for plotting.
u = reshape(u,n,n);

% Append on to u the boundary values from the Dirichlet condition.
u = [[feval(bfunc,x(1,1:n+2),y(1,1:n+2))];...
     [[feval(bfunc,x(2:n+1,1),y(2:n+1,1))] u ...
      [feval(bfunc,x(2:n+1,n+2),y(2:n+1,n+2))]];...
     [feval(bfunc,x(n+2,1:n+2),y(n+2,1:n+2))]];
```

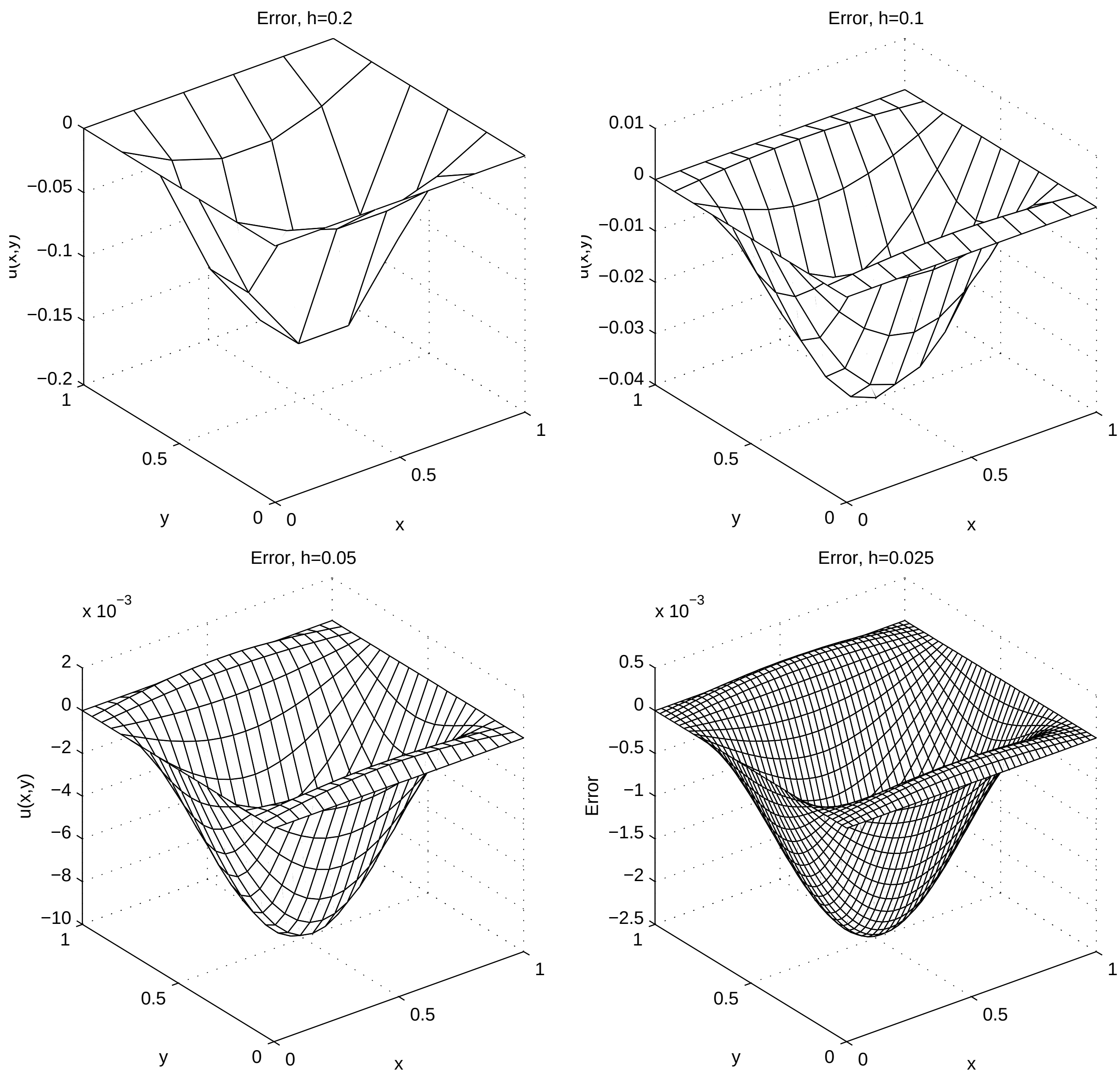
Example:

Use the above MATLAB code to solve the Poisson problem

$$\begin{aligned}
\nabla^2 u(x,y) &= -5\pi^2 \sin(\pi x)\cos(2\pi y) \ \text{ for } (x,y) \in \Omega = (0,1)\times(0,1) \\
u(x,y) &= \sin(\pi x)\cos(2\pi y) \ \text{ for } (x,y) \in \partial\Omega
\end{aligned}$$

```
function testFdPoisson

f = inline('-5*pi^2*sin(pi*x).*cos(2*pi*y)');
g = inline('sin(pi*x).*cos(2*pi*y)');

[u,x,y] = fd2poisson(f,g,0,1,39);
h = x(1,2) - x(1,1);

% Plot solution
figure, set(gcf,'DefaultAxesFontSize',8,'PaperPosition', [0 0 3.5 3.5]),
mesh(x,y,u), colormap([0 0 0]), xlabel('x'), ylabel('y'), zlabel('u(x,y)'),
title(strcat('Numerical Solution to Poisson Equation, h=',num2str(h)));

% Plot error
figure, set(gcf,'DefaultAxesFontSize',8,'PaperPosition', [0 0 3.5 3.5]),
mesh(x,y,u-g(x,y)), colormap([0 0 0]), xlabel('x'), ylabel('y'), zlabel('Error'),
title(strcat('Error, h=',num2str(h)));
```




Error, h=0.2
Error, h=0.1
Error, h=0.05
Error, h=0.025
u(x,y)
Error
x 10^{-3}
x
y
0
0.5
1
0.01
−0.01
−0.02
−0.03
−0.04
−0.05
−0.1
−0.15
−0.2
2
−2
−4
−6
−8
−10
−0.5
−1
−1.5
−2
−2.5