MATH466/462 Project 3. *Due in class on Wed, Feb 26, 2020*

Instruction: your project report should include necessary mathematical justification, description, and details of your algorithms/conclusions. Your MATLAB codes and generated outputs may be attached in the end of the report. Make sure you addressed all the questions in each problem. Both the report and codes will be graded. Please submit a printed hard-copy.

Problem A (20 pts): SOR Method and its optimal relaxation factor.

Classical iterative methods (e.g., Jacobi, Gauss-Seidel, SOR) are popular for solving a large-scale sparse linear system with a symmetric positive definite (SPD) matrix, but such methods can converge very slowly. SOR often outperforms Jacobi and Gauss-Seidel methods, if a good relaxation factor can be found, which is however highly depend on the system matrix structure.

1. Implement the SOR algorithm for SPD matrix in MATLAB as the following function :

```
function [x,iter]=SOR(A,b,omega,tol)
%Input: A is the system matrix
%       b is the right hand side vector
%       omega is the relaxation factor used in SOR
%       tol  is the stopping tolerance for relative residual reduction
%Output: x is solution of Ax=b, iter is the used iteration numbers

```

2. The SOR method with the optimal relaxation factor $\omega_{opt} \in (0, 2)$ shows a much faster convergence. Study (skip theory part) the model problem (3.19)-(3.20) given in
`http://www.math.umbc.edu/%7Ekogan/technical_papers/2007/Yang_Gobbert.pdf`
The following vectorized codes can generate the corresponding sparse linear system (3.19)-(3.20):

```
N=65;h=1/(N+1); %domain [0,1]*[0,1]
xx=h:h:1-h;[X,Y] = ndgrid(xx,xx); %interior uniform mesh
ffun=@(x,y) -2*pi^2*(cos(2*pi*x).*sin(pi*y).^2+sin(pi*x).^2.*cos(2*pi*y));
F=h^2*ffun(X,Y); b=F(:); %right-hand-side function values  as a column vector
A=gallery('poisson',N); %generate SPD matrix A in (3.19)
u=A\b; %solve by sparse direct solver, most accurate
U=reshape(u,N,N); mesh(X,Y,U) %reshape and plot the 2D solution u(x,y)

```

Write MATLAB script based on your above SOR code to generate a similar Figure 2 (see page 18 for more detail setting), which numerically verifies the given relaxation parameter $\omega_{opt} = \frac{2}{1+\sin(\pi h)}$ in (4.2) is indeed optimal (according to Theorem 11 ($d = 2$) in this paper).

3. (Bonus) Solve the same system with `pcg` solver. Plot in the same figure the residual norm convergence history of both SOR and `pcg` method. Which converges faster? You may take look the followig link on `pcg` solver:
`https://www.mathworks.com/help/matlab/ref/pcg.html`