

Name: _____

Date: _____

Collaborators: _____

(Collaborators submit their individually written assignments together)

Question:	1	2	3	4	5	6	Total
Points:	20	15	5	15	5	10	70
Score:							

Instructor/grader comments:

Comparing iterative algorithms for solving linear systems of equations

Compare the performance of Jacobi, Gauss-Seidel, Steepest Descent and Conjugate Gradient methods.

1. (20 points) Modify the code that you developed for Jacobi and Gauss-Seidel methods to use the same convergence test – the square of the 2-norm of the residue – that is used in the code for the steepest descent and conjugate gradients methods.
 - name your new functions `myjacobi2` and `mygauss_seidel2`. Keep the same calling and returning parameters.
 - Since the 'new' test doesn't require both the current and the previous version of the approximation to the solution, keep only the current one
 - Notice that you need the original matrix (A) and the right hand side vector (b) as well as 'iterative' matrix (T) and the vector (c).
 - Make sure you place the initial value of the square of the residue norm into your convergence array
2. (15 points) Use the code to conduct numerical experiments to compare the performance of the four methods when working with small full matrices.

Use the following parameters:

```
n = 30;  
A = genspdm(n);  
b = rand(n, 1);  
x0 = ones(n, 1);  
tol = 0.000000000001;  
maxit = 100;
```

The code for `genspdm()` is provided on the course website.

On the same graph plot the convergence parameter vs the iteration number for all methods. Place the code you wrote for this part of the homework in a single matlab file, e.g. **hw04a.m**

3. (5 points) Clearly describe your observations and conclusions in your project's readme file.
4. (15 points) Use the code to conduct numerical experiments to compare the performance of the four methods when working with mid-size sparse matrices.

Use the following parameters:

```
n = 1000;  
dens = 0.005;  
A = genspdm_sparse(n, dens);  
b = rand(n, 1);  
x0 = ones(n, 1);  
tol = 0.0000000000001;  
maxit = 100;
```

The code for `genspdm_sparse()` is provided on the course website.

On the same graph plot the convergence parameter vs the iteration number for all methods. Place the code you wrote for this part of the homework in a single matlab file, e.g. **hw04b.m**

5. (5 points) Clearly describe your observations and conclusions in your project's readme file.

Gitlab

6. (10 points) Create a gitlab project called **hw04** (name it exactly as shown). Upload **all** required matlab code and create your readme file. Share the project with the instructor.