Due Date:

1. Written Part Due on 11/02, Friday, in class.

2. Programming Part Due on 11/02, Friday, midnight. (Be aware that every problem involves implementation in Matlab).

**PART I: SENSITIVITY TO PERTURBATION**

We consider three types of systems

- Orthogonal systems
- Triangular systems
- Vandermonde systems

**PROBLEM 1.**

Find numerical solutions of the linear triangular systems with the $n \times n$ matrix $L$

\[
\ell_{ii} = 1, \quad i = 1 : n \\
\ell_{i+1,i} = -\beta_1, \quad i = 1 : n - 1 \\
\ell_{i+2,i} = -\beta_2, \quad i = 1 : n - 2 \\
\ell_{i,j} = 0, \quad \text{otherwise}
\]

and the following parameters, right hand sides and problem sizes

- $\beta_1 = 2.25$, $\beta_2 = -0.5$, and $\beta_1 = 0.8$, $\beta_2 = -0.16$,
- $b_1^T = (1/3, -2/3, 0, \cdots, 0)$, and $b_2^T = (1, -2, 0, \cdots, 0)$
- $n = 10, 30, 60, 120$

Let $x_1$ and $x_2$ be the solutions to the systems with $b_1$ and $b_2$, respectively.

1. Name a few familiar structures of the matrix. Is the matrix nonsingular? What are the eigenvalues of the matrix?

2. Provide a direct method (M1) and its matlab implementation to compute the solution, and compute the residual in higher precision (e.g. using vpa in matlab). Provide algorithm complexity in the document in terms of arithmetic operations and memory requirement.

3. What is the relationship between $x_1$ and $x_2$ in exact computation? Do your computed solutions satisfy the expected relationship? If not, find two different ways to characterize and explain the discrepancy.
4. Specify a condition on $\beta_i$, $i = 1, 2$, under which M1 is not safe to use with arbitrary right hand side.

5. Plot the singular values for each matrix.

**Problem 2.**

1. Describe two computational problems where the resolution of Vandermonde systems is involved in.

2. Give a case where a Vandermonde matrix is unitary, up to scaling.

3. Use the following code to generate real Vandermonde systems and observe their numerical behaviors

   ```
   n = 25, 50, 100
   v = linspace(0,1,n); A = vander(v);
   b = rand(n,1);
   ```

   Provide at least two algorithms by using matlab built-in routines, compute the residuals, and record your observations about the computed results.

4. Use matlab function `cond(A)` to find out the condition numbers of the matrices, and find the change as $n$ changes.

5. Recommend one of the algorithms you have used, and provide some explanations.

**Part II. Least Squares Problems and Orthogonal Systems**

Compute and compare the exact Fourier coefficients and the ones obtained numerically by using discrete Fourier transforms with the degree of trigonometric polynomials as $n = 50$, 500. Design your own presentation format on the comparison and offer your observations.

1. $f(x) = \sum_{k=1}^{5} \cos(3kx + 1/k) , x \in [0,2\pi]$

2. $f(x) = \alpha x^2 + \beta x$, $\alpha = 3$, $\beta = 7$, $x \in [0,2\pi]$. Cf. Homework-4.
Part III. Iterative methods

Consider numerical solution $u_{ij}, i = 0 : n, j = 0 : m$, on equally spaced grid to the discretized 2-dimensional Poisson equation (via finite difference) with

$$f(x, y) = xe^y, \quad (x, y) \in [0, 2] \times [0, 1],$$

and the boundary conditions

$$u(0, y) = 0, \quad u(2, y) = 2e^y, \quad y \in [0, 1],$$
$$u(x, 0) = x, \quad u(x, 1) = e^1 x, \quad x \in [0, 2],$$

for $n = m = 25, 50, 500$.

- Present the linear equations (the matrix and the right hand side). Find the condition numbers for the cases $n = 25, 50$.

- Use and implement in matlab two different iterative methods for the numerical solutions, describing the methods in documentation, including stop criteria and computation complexity per iteration.

- Plot the initial, one intermediate and the final computational results of each method.

- Illustrate experimental comparisons of the two methods.

- Try to use a direct method to compute the numerical solutions, and record your observations.