## Lab Session 5

**MA-571 :** Numerical Linear Algebra Lab 2022 R. Alam

---

### Errors and backward stability

1. Your task is to generate test matrices having pre-specified condition numbers. Write a matlab function to generate an $n$-by-$n$ matrix with 2-norm condition number `ka`:

   ```
   function A = matgen(n, ka)
   ```

   To obtain such a matrix proceed as follows. Choose random orthogonal matrices $U$ and $V$ from QR factorization of random matrices: `[U, R] = qr(rand(n))` and `[V,R] = qr(rand(n))`. Generate a diagonal matrix $D = \mathrm{diag}(d_i)$ with $d_i :=$ `ka`$^{-(i-1)/(n-1)}$ and then set $A := UDV^*$. Then the 2-norm condition number of $A$ is equal to `ka`.

   Check that the matrix $A$ generated by your function `matgen` has the condition number `ka` (call MATLAB function `cond(A)` to compute the condition number of $A$).

2. Test the stability of algorithms specified below and analyze the accuracy of solutions returned by them. For this purpose, proceed as follows:

   1. For each `ka` $= 10^4, 10^8, 10^{12}, 10^{16}$, use your `matgen` function to generate a random matrix $A$ of size $n = 100$ with condition number `ka`.
   2. Compute `b = A*ones(n,1).` Then `x =ones(n,1)` is the solution of $Ax = b$.

   Now consider the following algorithms:

   1. `ALG1:` Solve $Ax = b$ using backslash command (`GEPP`)
   2. `ALG2:` Solve $Ax = b$ using `GECP`
   3. `ALG3:` Solve $Ax = b$ using `GENP`
   4. `ALG4:` Solve $Ax = b$ by $A^{-1}b$ (MATLAB command `inv` computes inverse of a matrix).

   For each of the above algorithms, prepare a table giving values of the following quantities:

   `ka, bkerr, err, errbd`

   where

   - `bkerr` - is the backward error ($= \|Ax - b\|_2/\|A\|_2\|x\|_2$)
   - `err` - is the relative error in the solution
   - `errbd` - is the relative error bound of the solution. [ If $r = b - Ax$ then $(A+E)x = b$ where $E = rx^*/\|x\|_2^2$. Now for error bound, invoke perturbation theory.]

   Based on your results comment on the backward stability of `ALG1`, `ALG2`, `ALG3`, `ALG4`, and the accuracy of the solutions returned by these methods. Use the MATLAB command `cputime` (type `help cputime` for more information) to determine the time taken by each of these methods to solve the problems.

**********End**********