MATH 4093/5093 **Homework 5** **Due Fri, 3/4/2011**

Problem 1. Consider the matrix

$$A = \begin{pmatrix} 4 & -2 \\ 1 & 1 \end{pmatrix}$$

Compute by hand $\|A\|_1$, $\|A\|_2$, and $\|A\|_\infty$. Please write your calculations in detail, especially for $\|A\|_2$.
Hint: We derived the expression for $\|A\|_\infty$ in class, for $\|A\|_2$ use the Theorem on page 178 of the book, and the expression for $\|A\|_1$ is given in Exercise 7 on page 181 of the book (you do not need to derive it).

Problem 2. Consider the norms $\| \|_1$, $\| \|_2$, and $\| \|_\infty$ on $\mathbb{R}^n$.

(a) Prove that the norms $\| \|_2$ and $\| \|_\infty$ are equivalent.

(b) In class we proved that the norms $\| \|_1$ and $\| \|_\infty$ are equivalent. Use this fact together with the fact proved in part (a) to show that the norms $\| \|_1$ and $\| \|_2$ are equivalent. You have to use only these two facts *only*, without any additional calculations!

Problem 3. Many theorems that hold in finite-dimensional spaces are not true in infinite-dimensional spaces. One can think of an infinite-dimensional space as a space of infinite sequences: $\mathbf{u} = (u_1, u_2, u_3, \ldots)$, where u_j are real numbers ($j \in \mathbb{N} = \{1, 2, 3, \ldots\}$). In this space we can define the ℓ_1, ℓ_2 and ℓ_∞ norms as follows:

$$\|\mathbf{u}\|_1 := \sum_{j\in\mathbb{N}} |u_j| \ , \qquad \|\mathbf{u}\|_2 := \left(\sum_{j\in\mathbb{N}} |u_j|^2 \right)^{1/2} \ , \qquad \|\mathbf{u}\|_\infty := \sup_{j\in\mathbb{N}} |u_j|$$

(if $\{a_j\}_{j\in\mathbb{N}}$ is a sequence or numbers, then $\sup\limits_{j\in\mathbb{N}} a_j$ is defined as the the smallest number a such that $a_j \le a$ for all $j \in \mathbb{N}$).

(a) Give an explicit example of a sequence $\mathbf{u}$ such that $\|\mathbf{u}\|_\infty < \infty$, but $\|\mathbf{u}\|_1$ is infinite.

(b) Give an explicit example of a sequence $\mathbf{u}$ such that $\|\mathbf{u}\|_\infty < \infty$, but $\|\mathbf{u}\|_2$ is infinite.

(c) **Only for the students taking the class as MATH 5093!**
Give an explicit example of a sequence $\mathbf{u}$ such that $\|\mathbf{u}\|_2 < \infty$, but $\|\mathbf{u}\|_1$ is infinite.

(d) **Only for the students taking the class as MATH 5093!**
Explain simply why there cannot exist a sequence $\mathbf{u}$ such that $\|\mathbf{u}\|_1 < \infty$, but $\|\mathbf{u}\|_2$ is infinite.

Problem 4. Let

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} .$$

It is easy to check (and you do *not* need to do it!) that the matrix A can be written as $A = L_1 U_1$, as well as $A = L_2 U_2$, where

$$L_1 = \begin{pmatrix} 1 & 0 \\ 3 & 1 \end{pmatrix}, \quad U_1 = \begin{pmatrix} 1 & 2 \\ 0 & -2 \end{pmatrix}; \qquad L_2 = \begin{pmatrix} 1 & 0 \\ 3 & -2 \end{pmatrix}, \quad U_2 = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix}.$$

(a) In class we discussed that the arbitrariness in the choice of matrices L and U in $A = LU$ is in the choice of a non-singular diagonal matrix D such that $L_1 = L_2 D$ and $U_1 = D^{-1} U_2$. Find explicitly the matrix D for the pairs (L_1, U_1) and (L_2, U_2) given in part (a).

(b) Use the pair (L_1, U_1) from part (a) to solve the system $A\mathbf{x} = (\ 4 \ \ 6\ )^T$.

(c) Let

$$B = \begin{pmatrix} 3 & 4 \\ 1 & 2 \end{pmatrix} .$$

Construct a permutation matrix P, a lower triangular matrix L, and an upper triangular matrix U, such that

$$PB = LU \ .$$

Hint: You can do this with almost no additional calculations if you look carefully at the matrices A and B.

Problem 5. Let

$$A = \begin{pmatrix} 2 & 7 & 5 \\ 6 & 20 & 10 \\ 4 & 3 & 0 \end{pmatrix} .$$

(a) It is easy to show by direct computation that

$$\begin{pmatrix} 1 & 0 & 0 \\ \alpha & 1 & 0 \\ \beta & 0 & 1 \end{pmatrix} \begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 + \alpha a_1 & b_2 + \alpha a_2 & b_3 + \alpha a_3 \\ c_1 + \beta a_1 & c_2 + \beta a_2 & c_3 + \beta a_3 \end{pmatrix}$$

(you do *not* have to do this!). Use this fact to find a matrix

$$M_1 = \begin{pmatrix} 1 & 0 & 0 \\ \alpha & 1 & 0 \\ \beta & 0 & 1 \end{pmatrix}$$

so that, for the matrix A above,

$$M_1 A = \begin{pmatrix} * & * & * \\ 0 & * & * \\ 0 & * & * \end{pmatrix} ;$$

hereafter, the stars represent arbitrary numbers.

(b) Again, it is easy to check that

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \gamma & 1 \end{pmatrix} \begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{pmatrix} = \begin{pmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 + \gamma b_1 & c_2 + \gamma b_2 & c_3 + \gamma b_3 \end{pmatrix}$$

(you can use this fact without proving it). Use this to find a matrix

$$M_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \gamma & 1 \end{pmatrix}$$

so that for the matrix A above

$$M_2 M_1 A = \begin{pmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \end{pmatrix} .$$

(c) Show that

$$\begin{pmatrix} 1 & 0 & 0 \\ \alpha & 1 & 0 \\ \beta & 0 & 1 \end{pmatrix}^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ -\alpha & 1 & 0 \\ -\beta & 0 & 1 \end{pmatrix} , \qquad \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \gamma & 1 \end{pmatrix}^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -\gamma & 1 \end{pmatrix} .$$

(d) Use your results from the previous parts of this problem to construct an explicit LU decomposition of the matrix A.

Problem 6. Newton's method is a very efficient iterative method for finding the root of the algebraic equation $f(x) = 0$ (it also works for systems of equations). The idea of the method is represented simply by the picture on page 96 of the book. The procedure is the following: choose some number p_0 that you suspect is close to some root p^* of the equation $f(x) = 0$; let $\{p_n\}_{n=0}^{\infty}$ be the sequence defined by

$$p_{n+1} = p_n - \frac{f(p_n)}{f'(p_n)} .$$

If the sequence $\{p_n\}$ converges, then its limit, $\lim_{n\to\infty} p_n$, is a root of $f(x) = 0$.

The MATLAB code `newton.m` from the class web-site solves a single equation $f(x) = 0$ by using Newton's method. You will solve the equation

$$f(x) = x^3 + 3x^2 - 4x - 12 = 0 \ ,$$

whose roots are -3, -2, and 2. Write functions `fun.m` and `funder.m` returning the values of $f(x)$ and $f'(x)$, respectively. If you run the code `newton.m` as follows:

```
newton( @fun, @funder, 0.1, 1e-8, 100, 1 )
```

the code will perform Newton's iteration starting from $p_0 = 0.1$ and will iterate until the distance between the iterates p_{n-1} and p_n is no more than 10^{-8}; since the last argument of the function `newton` is 1, the function prints all intermediate steps (not just the final result). At each step of the Newton's iteration, the code prints out the number n, the value p_n, the difference $p_n - p_{n-1}$, and $\log_{10} |p_n - p_{n-1}|$ (which is approximately equal to the number of correct digits in the answer). The symbol `@` is used to create a *handle* to a function – on page 75 of Overman's *MATLAB Overview* you can find more about this; on pages 74–75 of the *Overview* you can read about the MATLAB command `feval` (used in `newton.m`).

For this problem, you have to attach only:

(a) printouts of your functions `fun.m` and `funder.m`;

(b) a printout of the output of running `newton.m` with accuracy 10^{-14}, and initial values p_0 equal to -0.1, -2.45, and 5.0.