Math 481 HOMEWORK 1 due Thursday, Feb 12, 2009

Fritz Keinert

For each problem that uses Matlab, you should hand in a printout of the relevant script or function file(s), or a transcript of your interactive session (use the `diary` feature), plus whatever outputs or plots are requested. Put the problems in the proper order, and label all printouts clearly. The final output should have full accuracy (`format long`); intermediate results can be shorter, if you want.

1. Use Matlab to produce graphs of the functions $y = 3\sin(\pi x)$ and $y = e^{-0.2x}$ in the same picture, for $x = 0 : 0.02 : 4$. Label both axes and the graph as a whole. Use `gtext` to label one of the intersection points of the graphs. (10)

2. **(a)** Write a Matlab function `signum` which does the same thing as the built-in `sign` function, but without calling `sign`.

Your function must be able to handle vectors and matrices as input, but try to write it without loops. You need to get by with a single statement, otherwise you can't do part (b).

Hint: Investigate the `find` function, as well as the meaning of `A>0`, where `A` is a matrix.

Test your routine by calculating `signum([-1,0;1,2])`.

(b) Rewrite the function from (a) as an inline function, and test it again. (7+3)

3. **(a)** Polynomials are represented in Matlab by their coefficient vectors (highest power first).

For the polynomials

$$\begin{aligned} p(x) &= 5x^4 + 3x^3 - x^2 + 1 \\ q(x) &= x^2 - x + 2 \end{aligned}$$

use Matlab's polynomial functions (`help polyfun`) to compute the following:

- $p(x) \cdot q(x)$
- $p(x)/q(x)$ (as polynomial part + remainder)
- $p'(x)$

This will only take one line each.

(b) Write a function `polyadd` which adds two polynomials, and use it to add $p(x) + q(x)$. (10)

4. For

$$A = \begin{pmatrix} 3 & 1 & 4 & 1 \\ 5 & 9 & 2 & 6 \\ 5 & 3 & 5 & 8 \\ 9 & 7 & 9 & 3 \end{pmatrix}, \qquad b = \begin{pmatrix} 2 \\ 7 \\ 1 \\ 8 \end{pmatrix},$$

calculate the following, using Matlab built-in functions:

determinant of A, 1-norm of b, solution of $Ax = b$, inverse of A. (10)

5. The number π is the circumference of a circle of diameter 1. One way to estimate it (due to Archimedes) is to inscribe a square, octagon, 16-gon, etc. into the circle and compute the circumference of each polygon. If p_n is the circumference of the regular inscribed polygon with 2^n sides, these numbers can be calculated recursively by

$$p_{n+1} = 2^n \sqrt{2\left(1 - \sqrt{1 - (p_n/2^n)^2}\right)}$$

starting from $p_2 = 2\sqrt{2}$. p_n will converge to π as n increases. This calculation is numerically unstable.

(a) Explain the source of the numerical stability problems.

A better way to calculate p_n is to calculate the numbers r_n first, which are defined as

$$r_n = 2\left(1 - \sqrt{1 - (p_{n-1}/2^{n-1})^2}\right),$$

so that

$$p_{n+1} = 2^n \sqrt{r_{n+1}}.$$

(b) By algebraic manipulation of the above formulas, show that the r_n can be calculated recursively by

$$r_{n+1} = \frac{r_n}{2 + \sqrt{4 - r_n}}$$

This calculation will be numerically stable.

(c) Make a table

n	p n by first formula	r n	p n by second formula
3	xxx	xxx	xxx
4	xxx	xxx	xxx
5	xxx	xxx	xxx
...			
60	xxx	xxx	xxx

The first column of p_n will converge for a while, then turn into garbage. The other two columns will be accurate.

(d) Eventually, the second method will also fail on any machine that uses IEEE arithmetic. Explain why. (10)