Homework #1  
Math 567, Fall 2016  
Due Friday Sept. 23th

1. Find the leading term in the truncation error $\tau$ of the following finite difference approximation

$$\tilde{u}(x) = \frac{1}{h} \int_{x-h/2}^{x+h/2} u(\xi) d\xi \approx u(x) + \tau$$

Hint: Expand $u(\xi)$ about $x$ and integrate the terms in the resulting Taylor series. Integrate the terms in the expansion with as little algebra as possible!

2. In finite volume schemes, we only have a cell average values $\tilde{u}(x)$, rather than the pointwise values $u(x)$. What is the truncation error in the following approximation to $u''(x)$?

$$\frac{\tilde{u}(x+h) - 2\tilde{u}(x) + \tilde{u}(x-h)}{h^2} \approx u''(x) + \tau.$$  

Assume that you know $\tilde{u}(x)$ exactly, so you can use Problem 1 to write $u(x) = \tilde{u}(x) - \tau$.

3. Search for “finite difference coefficient” in Wikipedia to find the coefficients for a centered fourth order approximation to $u''(x)$. Compute the leading order term in the truncation error. Use as little algebra as possible!

4. As you saw in 2 above, replacing pointwise values with averages didn’t change the order of accuracy of the second order approximation to $u''(x)$. But suppose you want to use the scheme you found in (3), above, but only have cell averages, not point-wise values. How can you get a fourth order approximation to $u''(x)$ using only cell averages? Hint: Find a correction term $C$ to the average that allows you to replace $\tilde{u}(x)$ with $u(x)$ and preserve fourth order accuracy. That is,

$$u(x) = \tilde{u}(x) + C + O(h^4).$$

The value $C$ should be expressed as a finite difference approximation, based on values $\tilde{u}(x-h)$, $\tilde{u}(x)$ and $\tilde{u}(x+h)$. Draw a sketch so you are clear about where your point-wise values are evaluated, and where the averages are taken. Hint: Use Problem 1.

5. Higher order finite difference approximations used in solving hyperbolic equations can involve computing a “one-sided approximation” to the first derivative of a function. Given give values $u(x_0)$, $u(x_1)$, $u(x_2)$, $u(x_3)$, $u(x_4)$ for equally spaced points $x_j = (j-0.5)h, j = 0, 1, 2, 3, 4$, derive a fourth order finite difference approximation to $u'(0)$.

- Sketch the location of the equally spaced points.
• Sketch the location of where the approximation should be made (i.e. at \( x = 0 \)).
• Write down a Taylor series expansion for each value of \( u(x_j) \) about the point \( x = 0 \).
• Describe the linear system you would need to solve to construct the stencil for the fourth order approximation to \( u'(x) \).
• Solve the linear system to get the coefficients for the approximation. Ideally, you can obtain the exact rational expressions for the coefficients. Hint : Use the “rat” function in Matlab.
• Verify numerically that your approximation is fourth order. Hint : You can show that it gives the exact derivative of the functions \( u(x) = x^n \) for \( n = 0, 1, 2, \ldots, 4 \).
• Use your approximation to approximate the derivative of \( u(x) = \sin(x) \) at \( x = 0 \) for a sequence of \( h \) values, e.g. \( h = 1, h = 0.5, h = 0.25, \ldots, 2^{-N} \) for some large \( N \) (maybe \( N = 8 ? \)). Show that your approximation approaches the expected convergence rate using the numerical approach you used in Homework #0.

6. Using the functions \texttt{fdcoeffF.m} or \texttt{fdcoeffV.m} from the course website, write a function that takes as input a derivative \( k \) plus some other numerical parameters and returns an approximation to the derivative of that function, to the requested order of accuracy. Verify that you are getting the correct order of accuracy by doing a grid convergence study of your results using the functions \( u(x) = \sin(x) \) and \( u(x) = J_0(x) \), where \( J_\nu(x) \) is the Bessel function of the first kind. Test your code for \( (k,p) \) combinations \((0,2), (1,2), (1,4), (2,2), (2,4), (4,2) \) and \((4,4)\).

   Use \texttt{spdiags} to construct a banded matrix of your coefficients so you can construct the derivatives by multiplying by this banded matrix. Your code for the approximations should not require any loops.

   Turn in your convergence rates for the \( (k,p) \) given above, and a few representative loglog plots showing that your errors converge to 0, up to a point.

   Answer this question. Why don’t get get good convergence beyond about \( 1 \times 10^{-6} \) for the fourth order derivatives?

   You can use the sample code on the course website to get started.

7. Prove that the eigenvalues given in Equation 2.23 of your book are correct.

8. Derive the inverse of the matrix \( A \) given in Section 6.3.2.