Math 4650 Final Review Topics

- 5.1: Understand the definition of a "well-posed problem": existence, uniqueness, and stability. Most ordinary differential equations are well-posed, but understand that some may not be (for example, $y' = y/x$ with $y(0) = 1$).

- 5.2: Know Euler's method. Be able to use it in a simple case. You don't have to know how to prove the general convergence estimate

$$|y(t_i) - w_i| \le \frac{hM}{2L}\left[e^{L(t_i-a)} - 1\right],$$

  but understand what it means and why it's useful.

- 5.3: Understand higher-order Taylor methods. Be able to derive a second-order Taylor method and use it in a simple case. Understand how Taylor methods are the basis for local truncation error estimates.

- 5.4: Understand Runge-Kutta methods. Know the basic idea behind their derivation (trying to approximate the Taylor methods without actually computing derivatives of the function). Know the three basic second-order methods (midpoint, modified Euler, and Heun) and the basic fourth-order method (FORK), and be able to use the second-order methods in a simple case. Understand why Runge-Kutta methods are complicated to derive, and why the number of function evaluations is not always the same as the order of the method. Be able to derive the local truncation error of a second-order method.

- 5.5: You don't need to know the algorithm for Runge-Kutta-Fehlberg, but understand the basic idea behind why it works. (Use of a higher-order method to estimate the error, and use of the estimated error to modify the step size.)

- 5.6: Understand the basic idea behind Adams methods, and how they are fundamentally different from Runge-Kutta or Taylor methods. Know the basic principle of getting an Adams method started. Know the difference between explicit (Adams-Bashforth) and implicit (Adams-Moulton) methods. Understand why an n-step Adams-Bashforth method has local truncation error $O(h^n)$, while an n-step Adams-Moulton method has local truncation error $O(h^{n+1})$. Be able to use a 2-step Adams-Bashforth or a 1-step Adams-Moulton method in simple cases. Know how to combine Adams-Bashforth and Adams-Moulton method into a predictor-corrector method.

- 5.7: Understand the basic idea behind adaptive predictor-corrector methods (use two methods with the same order of error and a known coefficient to get the actual value, and from this to estimate the actual error of the better method).

- 5.10: Be able to solve a difference equation. Know how to set up the difference equation for a multistep method, and know exactly what the stability conditions are for the roots of its characteristic polynomial (and where these conditions come from). Know deep in your heart the fundamental theorem of numerical differential equations: Consistency plus zero-stability equals convergence. (Of course, know what all these words mean.)

- 5.11: Understand what the region of absolute stability is, how to compute it in simple cases (as on the homework), and how to deduce A-stability from it. Understand the motivation behind the definition of A-stable: if the actual differential equation has decaying solutions, the difference scheme should also have decaying solutions.

- General Chapter 5: Be able to compute local truncation errors by using Taylor expansions. Know how local truncation error relates to global error (as made explicit with the Euler formula). Understand the advantage of Runge-Kutta methods (higher accuracy, as on the homework), and the advantage of Adams methods (fewer function evaluations, if you set it up correctly). Be able to explain these things in words.

- 6.1: Understand the basic Gaussian elimination with back-substitution algorithm. Know how to count operations in such an algorithm, and why multiplications/divisions are counted separately from additions/subtractions. Know why Gaussian elimination is better than Gauss-Jordan elimination. Be able to do Gaussian elimination in simple cases.

- 6.2: Understand why some kind of partial pivoting must be used in order to avoid roundoff errors. Know the difference between partial pivoting, scaled partial pivoting, and complete pivoting. Understand why a correctly-done pivoting does not add significantly to the algorithm length, but greatly improves accuracy. Be able to do Gaussian elimination with partial or scaled partial pivoting in simple cases.

- 6.4: Know how to compute determinants, and why one almost never computes determinants using row- or column-expansions. Be able to compute determinants using Gaussian elimination to reduce to upper triangular matrices.

- 6.5: Understand why a factorization $A = LU$ is important and useful. Be able to solve $LUx = b$ using the decomposition $Ly = b$ and $Ux = y$. Understand how to get a factorization using Gaussian elimination, and be able to do it in simple cases. Understand why a matrix might not have an LU factorization.

- 6.6: Know the definitions of strict diagonal dominance, symmetry, positive-definiteness, and band matrix. Know what each of these properties would tell you about factorizations, and be able to do Cholesky factorization in simple cases. Understand Crout's algorithm for tridiagonal matrices, and why it works so much better than Gaussian elimination on such matrices. Understand generally why one should always use specialized algorithms for band matrices rather than Gaussian elimination.