MATH 310: Numerical Analysis
Details for the final exam

The final exam in MATH 310 will be Saturday, December 19 from 8:00 - 10:00 in our regular classroom. You should bring a calculator to perform rudimentary computations, but you are not permitted to access symbolic computations or the internet on your calculator. Any student violating this rule will receive an 'F' on the exam.

The exam is cumulative and you should expect to be asked to write or read a little bit of code and to perform a few iterations of the algorithms we covered.

You may bring a 'cheat sheet' to the exam with the following, *and only the following*, contents:

1. Details of complete and natural B-splines. You can be quite thorough here, including the matrices you need, the equations for the auxiliary coeffiecients c_i, and the values of $B_i(x)$, $B_i'(x)$, $B_i''(x)$ at various nodes.
2. Formulas for the trapezoid rule, composite trapezoid rule, corrected trapezoid rule, Simpson's rule, composite Simpson's rule, Midpoint rule, and composite Midpoint rule.
3. Formulas for error in polynomial interpolation, bisection/secant/Newton's method for root finding, I'll provide a cheat sheet on the test with error estimates for you to consult that does *not* include the ones listed in the previous sentence.

This 'cheat sheet' will be collected with the final exam.

Topics emphasized are the same ones covered on the first and second midterm, but you do not need to memorize the error terms *except for Taylor's Theorem with Remainder (1.3) in the text*.

- Know by heart the following theorems: Taylor's Theorem with remainder, version (1.3) in text, Theorem 3.1 on Bisection Convergence and Error, Theorem 3.2 on the Newton Error Formula.
- Know and be able to use the following definitions: $\mathcal{O}$ notation, absolute error, relative error, machine epsilon, order of convergence of $x_n \to \alpha$.
- Understand floating point arithmetic, including its pitfalls, the concept of machine epsilon, and overflow and underflow errors.
- Know and be able to use the following algorithms: Horner's method, difference approximations to the derivative $f'(x)$, root finding algorithms including bisection/secant/Newton's method, Durand-Kerner, Lagrange interpolant, Newton interpolant, divided differences, piecewise polynomial interpolation, B-splines, and numerical integration quadrature rules.

  "Knowing" includes understanding the algorithms' strengths and weaknesses, and their errors, and rates of convergence.
- Be able to use Gaussian elimination to solve a linear system of three equations in three unknowns.
- Examples: for instance, Chebychev nodes, Runge example, ...

There is no guarantee that the list above is complete, but it gives you the idea of what you should focus your studying on.