The exam will cover all material taught from Chapters 2, 4 and 5 as well as Chapter 7 sections 1-2.

Here are some ideas of things you should be able to do on the exam. Not all these topics will be on the exam. If a topic from the covered material isn't here, that doesn't mean it won't be on the exam. But for the most part, the exam questions cover a subset of this list.

You should also look over your homework problems for ideas of things to study. If I assigned a question, I did so because I thought the problem was important. If I wrote the question myself, I must have thought it was especially important!

The exam is closed book, with no notes. No calculators will be allowed. It will be just you and your wits and a pencil.

- We have seen a number of numerical methods so far in the class. Here are a few.
  - Bisection.
  - Newton's method for finding roots.
  - Secant method for finding roots.
  - Seeking fixed points by iteration.
  - LU factorization.
  - LU factorization with partial pivoting.
  - Solving upper or lower triangular systems via backwards and forwards substitution.

  For each of these algorithms, you should be able to perform the first few steps of each algorithm in a simple case to show that you understand the steps of the algorithm.
- Show that you can write a little matlab code to implement a simple algorithm. For example, you should be able to write a basic function that implements Newton's method.
- Understand the IEEE 754 standard. In particular, given a baby version of the standard (8 bits, say with 1 sign bit, 3 exponent bits, and 4 mantissa bits) be able to answer question such as:
  - Given a bit representation, what number does it correspond to?
  - How many different numbers can be represented?
  - What is machine epsilon?
  - What is the smallest positive normalized number that can be represented? What is the largest?
  - How many non-normalized numbers are there?
  - If asked to add two numbers with a rounding mode, what is the resulting floating point number?
  - What bit patterns represent zero? Infinity? NaN?

  - Given a floating point number, what is the next largest number that can be represented? How is this answer related to machine epsilon?

- Given a matrix/vector computation, be able to compute the number of floating point operations required. See, e.g. Problem 7.6.
- We have used Taylor's theorem with remainder on a number of occasions. **Know the statement of this theorem**. Understand how a Taylor polynomial and the remainder term work together. In particular, be able to use the remainder term to estimate error. See, e.g. homework 3 problem 1.
- Be able to **state the Intermediate Value Theorem**. Why is the intermediate value theorem important for the bisection method?
- What is the difference between absolute and relative error? Given a true value and an approximation be able to compute both of these quantities.
- What is the difference between linear convergence and quadratic convergence? What is the definition of convergence of order δ?