Review Guide for Final Exam on Tuesday, 10 December 2019, 10:15am–12:15

This exam is in-class, closed-book, and 120 minutes long. No calculators or computers or phones are allowed. However, **you may bring 1/2 of one sheet of letter size paper** with any notes you want on it. This exam covers the whole course, but material from least squares (section 7.6) onward will be emphasized; earlier material appeared on the Midterm.

I encourage you to work through this Guide with other students. Read the relevant parts of the textbook (Greenbaum & Chartier, 2012) and identify the first items you *don't* understand as you get to them. Talk about that! If you come prepared it will be easy.

How I will create the exam. I'll ask myself: "Did a question like that appear on homework or a worksheet?" and "Is it easy to do without a computer?" and "Will the answers be cluttered and hard to grade?" (Should be: yes, yes, no.)

Problems will be in these categories:

- apply an algorithm/method in a simple concrete case
  E.g. Do two steps of bisection on this problem.
- state a theorem or definition (no need to *prove* theorems)
  E.g. State Taylor's Theorem with remainder.
  E.g. Define the Chebyshev points on the interval $[-1, 1]$.
- write a short pseudocode or Matlab code to state an algorithm
  E.g. Write Gauss elimination as a Matlab *code or pseudocode.*
  E.g. State the midpoint rule for ODEs as a Matlab *code or pseudocode.*
- explain/show in words (write in complete sentences)
  E.g. Which of [these methods] is best for [this problem]? Explain.
- derive an algorithm
  E.g. Derive Newton's method. Also draw a sketch which illustrates one step.

Sections. See these textbook sections which we covered in lecture and homework:

2.1–2.11	introduction to Matlab
4.1–4.5	solving nonlinear equations in one variable
5.2–5.4	introduction to floating-point arithmetic
7.1–7.3,7.6	solving linear systems
8.1–8.6	polynomial interpolation
10.1–10.5	numerical integration
11.1–11.2	numerical solution of ODE IVPs

Also read the Chapter introductions for Chapters 2, 4, 5, 7, 10, and 11. You can omit Chapters 3, 6, 9, and 12–14 entirely, but rereading Chapter 1 is not a bad idea.

Slides to review. I have posted two sets of slides, good for reviewing particular material:

- `bueler.github.io/polybasics.pdf`: *How to put a polynomial through points*
- `bueler.github.io/M310F19/euler302.pdf`: *2 illustrations of Euler's method*

Definitions. Be able to use these words correctly and/or write a definition if requested.

- the *absolute error* of $\hat{y}$ versus the exact value y: $|\hat{y} - y|$
- the *relative error* of $\hat{y}$ versus the (nonzero) exact value y: $|\hat{y} - y|/|y|$
- *fixed point* and *fixed point iteration* (section 4.5)
- *normal equations* (section 7.6, p. 167, equation (7.14))
- *Chebyshev points on* $[-1, 1]$ (p. 192)
- *ordinary differential equation initial value problem* (ODE IVP) (p. 251)
- *local truncation error* for a one-step ODE scheme (p. 271)

Theorems. You should understand the statements of these theorems, and be able to apply them in particular cases. I will not ask you for the proofs.

- intermediate value theorem (Thm 4.1.1)
- Taylor's theorem with remainder (Thm 4.2.1) MEMORIZE
- Newton's method converges quadratically theorem (Thm 4.3.1)
- fixed point convergence theorem (Thm 4.5.1)
- polynomial interpolation error theorem with remainder (Thm 8.4.1) MEMORIZE
- piecewise-linear interpolation error estimate (p. 198, section 8.6)
- error formulas for trapezoid and Simpson's rules (Table 10.3 on p. 235)[1]
- theorem on order of accuracy of Gauss quadrature (Thm 10.3.1)
- existence and uniqueness theorem for ODE IVPs (Thm 11.1.2)

Algorithms. Recall or re-derive these algorithms as needed.

- bisection method (section 4.1)
- Newton's method (section 4.3)
- secant method (section 4.4.3)
- forward/backward substitution for triangular systems (section 7.2)
- Gaussian elimination *as LU decomposition* (section 7.2)
- Gaussian elimination *with partial pivoting* (subsection 7.2.3)
- normal equations for least squares problems (section 7.6)
- constructing the interpolating polynomial (sections 8.1,8.2):
  - Vandermonde matrix method
  - Lagrange's formula for the polynomial
- piecewise-linear interpolation (section 8.6)
- Newton-Cotes formulas: trapezoid, Simpson's (section 10.1)
- composite versions of above (section 10.2)
- Gauss quadrature (section 10.3)
- Clenshaw-Curtis quadrature (section 10.4)
- Euler method for ODE IVPs (section 11.2)
- midpoint method for ODE IVPs (section 11.2)
- trapezoid method (implicit) for ODE IVPs (section 11.2)
- classical Runge-Kutta (RK4) method for ODE IVPs (section 11.2)

[1] Table 10.3 will be printed on the exam.

The three key concerns. Your key concerns about the above algorithms should be:

(1) What problem does it solve?
(2) Can I run the algorithm by hand in small cases with nice/convenient numbers?
(3) How does it compare to the other algorithms which solve similar/same problems?

Other concepts.

- anonymous functions in MATLAB (section 2.8)
- number of steps k for bisection to reduce interval size to 2δ (section 4.1, p. 78)
- floating-point representation and IEEE double precision (sections 5.3 & 5.4); be able to *understand* a description of the bit representation and then *find* the machine precision ϵ, the largest representable number, and the smallest positive (normal) representable number
- row operations as left multiplication by lower-triangular matrices (section 7.2)
- counting operations (subsection 7.2.1; e.g. exercise #6 on p. 176)
- using a factorization $A = LU$ to solve $A\mathbf{x} = \mathbf{b}$ by two triangular solves (section 7.2)
- why the Chebyshev points are superior for polynomial interpolation? (see the remainder term in Theorem 8.4.1)
- what is the basic idea behind all numerical integration methods in Chapter 10? (replace the integrand by a polynomial and integrate that)
- why are Runge-Kutta methods more convenient than Taylor methods? (we only need to evaluate $f(t, y)$ and not its derivatives)