Master Syllabus
MTH361 Numerical Analysis
Cluster Requirement 1C (Intermediate Writing)

COURSE DESCRIPTION An introductory course in analysis of numerical methods widely used in science and engineering. Basic topics: root finding, curve fitting, matrix algebra, integration and differentiation, solutions of systems of linear and nonlinear ODEs, solutions of PDEs, optimization and introduction to parallel computing.

PURPOSE Scientific computing has become an indispensable research tool and is vitally important for studying a wide range of physical and social phenomena. Students will examine the mathematical foundations of well-established numerical techniques and explore their usage through practical examples drawn from various disciplines including engineering and applied mathematics as well as the life and physical sciences.

A major focus of this class will be to develop the skills needed for mathematical writing and communications. Students will be expected to read with comprehension and critically interpret technical mathematical writing including proofs and algorithms. In homework assignments, students will be expected to demonstrate clear and comprehensible mathematical writing, and develop competence in the conventions of formal mathematical proofs and rigorous arguments. Students will also be expected to demonstrate the ability to collect, present, and evaluate mathematical writing from textbooks, journal papers, and websites.

COURSE SCHEDULE

A. **Weeks 1-3 :** Root finding: bisection, fixed point iterations, Newton's method.

The first homework will be based on topic A.

B. **Weeks 4 & 5:** Curve fitting: Polynomial interpolation. Splines.

C. **Week 6:** Differentiation and integration: Simpson's rule. Gauss quadrature.

The second homework will be based on topics B and C.

D. **Weeks 7 & 8:** Solution of linear systems: direct and iterative solutions. Other techniques for solving systems of equations, e.g., the Krylov subspace methods.

E. **Week 9:** Numerical solution of ODEs: Solution of initial value ODEs: Euler and Runge-Kutta methods. Eigenvalue problems. Boundary value problems using the shooting method.

The third homework will be based on topics D and E.

F. **Week 10:** Solution of parabolic, hyperbolic, elliptic PDEs using the Finite-difference method. Uniform and non-uniform grids. Convergence, and stability.

G. **Week 11:** Matlab Optimization techniques: linear programming, simplex method for nonlinear problems.

The fourth homework will be based on topics F and G.

H. (If time permits) Introduction to high performance computing and parallel computing. Monte Carlo simulations. Multigrid algorithms.

I. (If time permits) Strong-forms and weak-forms of PDEs: weighted residual method, collocation method, least square method, and Galerkin method. Introduction to the finite-volume method.

Additional homework will be based on topics H and I as appropriate.

LEARNING OUTCOMES

Course-Specific Learning Outcomes: Upon course completion, students will be able to solve problems by developing their own algorithms as well as implementing efficient algorithms in Matlab. Problems that students are expected to formulate and solve include:

- Root finding; solutions for nonlinear algebraic equations
- Solution of systems of linear equations
- Interpolation and curve fitting methods
- Numerical differentiation and integration
- Numerical solution of ordinary differential equations
- Numerical solution of partial differential equations
- Numerical analysis such as accuracy, stability, and convergence
- Matlab numerical algorithms implementation
- Matlab optimization tools
- Parallel computing

University Studies Learning Outcomes: After completing this course, students will be able to

- Read with comprehension and critically interpret and evaluate written work in discipline-specific contexts.
- Demonstrate rhetorically effective, discipline-specific writing for appropriate audiences.
- Demonstrate, at an advanced level of competence, use of discipline-specific control of language, modes of development and formal conventions.
- Demonstrate intermediate information literacy skills by selecting, evaluating, integrating and documenting information gathered from multiple sources into discipline-specific writing.

Students are expected to develop clear and comprehensible technical writing in the mathematical sciences. Every homework assignment will be evaluated on the basis of its mathematical communication. Students will be provided with resources for developing technical writing skills in the mathematical sciences, and with critical feedback from the instructor about their writing.

COMPUTING LABORATORY

Students should be proficient in linear algebra, multivariate calculus, and are somewhat familiar with programming. The course will make extensive use of Matlab.

SAMPLE TEXTS & READING

Numerical Analysis by Burden & Faires and online sources (review papers and lecture slides)
Mathematical Writing by Knuth, Larrabe, and Roberts and other online sources.

GRADING

Written homework assignments and project(s) 60%

Grading will be based on mathematical accuracy (20%) and on writing (40%).

Midterm exam 15%

Final exam 25%

Intermediate Writing Course Criteria: This course will satisfy the following criteria:

- *The course will employ writing as a method of deepening student learning:* This course is, for many students, the first in which they are expected to master the writing of mathematical proofs, presentation of computational algorithms, and description of the numerical results. Students' understanding of the mathematical material is predicated on the ability to identify the formal conventions required in these three contexts. For this reason, effective discipline specific writing has long been a major goal in this class.
- *Faculty provide feedback, on-going guidance, and clear expectations for "effective" written response:* Homework assignments and projects required in this course will focus on the development of effective discipline specific writing. Relevant elements of mathematical writing will be explicitly discussed in class (using the book by Knuth *et al.* as well as other online resources) when homework or project is assigned. Once the assignment is returned with instructor's comments, students will be encouraged to work in groups to improve the assignment and submit a revision. After this revision is returned, students will have one more opportunity to address weaknesses in the writing in response to instructor's comments and peer reviews.
- *Writing accounts for 40-60% of the final grade:* As described above, the written homework assignments for this course account for 60% of the final grade. Of this, fully 2/3 of the grading (or 40% of the total grade) will depend on the students' writing. As described above, this is needed as the students' understanding is intertwined with their understanding of the writing forms, language, and conventions of this field.
- *Students must complete at least 20 pages of writing:* Each homework assignment/project typically requires 5-10 pages of writing, and a minimum of 4 assignments is typical for this course.

OUTCOME MAP:

Univ St Learning Outcome	Teaching and Learning Activities	Student Work Products
1. Read with comprehension and critically interpret and evaluate written work in discipline-specific contexts.	Lectures will explicitly address the process of writing proofs, presenting numerical algorithms, and discussing the results of computational experiments.	In each written assignment, at least one question will focus on presenting a proof, algorithm, or numerical results discussed in the reading.

2. Demonstrate rhetorically effective, discipline-specific writing for appropriate audiences.	Using classroom colleagues as the target audience, students will use the group review of written assignments to develop effective writing,	In peer review sessions, students will provide feedback on the clarity of presentation and the use of discipline-specific rhetorical devices as described in the book by Knuth et al.
3. Demonstrate, at an advanced level of competence, use of discipline-specific control of language, modes of development and formal conventions.	An iterative process that includes peer-review as well as instructor comments will help students develop effective use of discipline specific language and forms.	One the course of the semester, the students will produce written assignments that will demonstrate (1) presenting proof in mathematically clear language and appropriate logical conventions and formalisms, (2) Clear description of a numerical algorithm, and (3) Clear discussion and analysis of computational results. These will require an integration of various written materials, an evaluation of the clarity and mathematical rigor of the sources, and clear documentation (outcome #4) as well as a presentation that includes discipline specific language, mathematical formalisms, and logical structure (outcome #3).
4. Demonstrate intermediate information literacy skills by selecting, evaluating, integrating and documenting information gathered from multiple sources into discipline-specific writing.	The understanding of various proof techniques and presentation approaches, and algorithm descriptions and pseudocode writing will be developed through a combination of textbook, articles, and online resources and the instructor's lectures.	

Intermediate Writing Course homework sample:

The aim of each homework is to create a mathematical paper or book chapter explaining the topic (or topics) covered. This will be accomplished through a series of questions that form an outline to the paper.

For example, a typical first problem for topic A would be:

> "What are root-finding methods needed for? What types of root-finding methods are there? Describe each method and explain how do they differ from each other? What are the advantages and shortcomings of each method and what types of problems are best suited for each type of method?"

This question would be grade both on (33%) whether the answer is correct and complete from a factual and mathematical point of view and on (67%) the clarity of the exposition, the use of a clear verbal explanation closely connected with the mathematical formulas that describe the method. It is of particular importance that mathematical terms are carefully defined (for example, a clear understanding of terms such as "starting value", the "n-th iterate", the "convergence rate", or a "fixed point") and their connection with the symbolic notation needs to be made explicit.

This question would be followed by a series of 3-5 questions specific to the methods of interest: bisection, fixed point iterations, Newton iteration, etc. A typical question from this series would have the following parts:

- What is a fixed point?
- How is the fixed point of a function related to the problem of root finding?
- Under what conditions does a function have a fixed point? Are these conditions necessary or sufficient? Why?
- <u>Prove</u> that given these conditions the function has a fixed point.
- Under what conditions can we guarantee that the fixed point is unique? Prove that this is the case. What type of a proof technique did you use for this? Are these conditions necessary or sufficient? Why?
- Summarize the above conditions as a formal theorem.
- Under what conditions does the the fixed point algorithm converge? Prove this. What is the rate of convergence of this algorithm? What factors will this depend on?

Further questions will focus on writing a code and describing the results in a combination of words, graphs, and tables. The use of pseudo-code and algorithms within the document will be expected. All homework will be expected to use LaTex.