CS 3513: Numerical Methods for Digital Computers

Required Course: Required
Course Number: CS 3513
Course Name: Numerical Methods for Digital Computers
Credit Hours: 3
Lecture Hours: 3
Lab Hours: 0
Instructors: Dr. Thanh Thieu

Book Title(s): Elements of Numerical Computation
Book Author(s): J.P. Chandler
Book Year(s): 2010

Book Title(s): Numerical Mathematics and Computing, 7^{th} edition
Book Author(s): W.Cheney and D.Kincaid
Book Year(s): 2013

Book Title(s): Numerical Methods, 4th edition
Book Author(s): J D Faires and R L Burden
Book Year(s): 2012

Course Description: Errors, floating point numbers and operations, interpolation and approximation, solution of nonlinear equations and linear systems, condition and stability, acceleration methods, numerical differentiation and integration.

Course Prerequisites: MATH 2153 (Calculus II(A)); MATH 3013 (Linear Algebra (A)) or concurrent enrollment; or MATH 3263(Linear Algebra and Differential Equations) and knowledge of programming.

Course Goals:

- Understand different types of generated errors in computation
- Recognize machine representation of floating-point numbers and associated properties
- Apply Taylor expansion and approximate function derivatives
- Program iterative algorithm to solve non-linear equations
- Apply acceleration methods to increase convergence of iterative algorithms
- Analyze condition of a problem and stability of an algorithm
- Implement Gaussian elimination to solve systems of linear equations
- Understanding of interpolation methods and difficulty in extrapolation
- Calculate definite integral using numerical approximation

Student Outcomes:

Student Outcome	Course Outcome
1	• Identify segments of the input domain that ill-condition the problem or destabilize the algorithm • Transform the original problem formulation into a solvable form, including non-linear, linear, or Taylor approximation
2	• Implement algorithms that solve non-linear equations, system of linear equations, and convergence acceleration • Evaluate accuracy of the solution given the platform-specific numerical representation
3	• Follow programming guidelines to write code that communicates its purpose clearly and effectively, using comments and self-documenting variables, methods and organization
4	• Understand how generated errors in unstable algorithms can have dangerous or fatal consequences, and use transformation strategies to redesign the algorithm into a stable version
6	• Practice code reuse and encapsulation in implementation of root finding methods • Analyze the trade-offs between speed, resource use, and accuracy of the computed solution

---

Course Topics:

Knowledge areas that contain topics and learning outcomes covered in the course

Knowledge Area	Total Hours of Coverage
Algorithms and Complexity (AL)	2
Architecture and Organization (AR)	3
Computational Science (CN)	27
Parallel and Distributed Computing (PD)	1.5
Programming Languages (PL)	1
Software Engineering (SE)	2
Social Issues and Professional Practice (SP)	0.5

Body of Knowledge coverage

Knowledge Area	**Knowledge Unit**	**Topics**	**Hours of coverage**
AL	Basic Analysis	Empirical measurements of performance, Time and space trade-offs in algorithms	2
AR	Machine Level Representation of Data	Fixed- and floating-point systems	3
CN	Processing	Numerical methods	10
CN	Numerical Analysis	Error, stability, convergence, function approximation including Taylors series, interpolation, extrapolation, and regression, Numerical differentiation and integration, Differential equations	17
PD	Parallel Decomposition	Independence and partitioning, Data-parallel decomposition	0.5
PD	Parallel Algorithms, Analysis, and programming	Parallel matrix computations	1
PL	Advanced programming constructs	Lazy evaluation, language support for checking assertions, invariants, and pre/post-conditions	1
SDF	Algorithms and Design	Problem-solving strategies, iterative and recursive numerical algorithms, divide and conquer strategies	2
SDF	Development methods	Testing fundamentals and test case generation	1
SE	Tools and Environment	Version control systems	1
SP	History	History of numerical computation algorithms	0.5