

CS4390 & CS5390
Applied Numerical Computing for Multimedia Applications
Spring 2024
Syllabus

Title:	CS4390: Applied Numerical Computing for Multimedia Applications, CRN 24695 crosslisted as CS5390, CRN 27258
Term:	Spring 2024
Duration:	Tue 01/16/2024 through Thu 05/02/2024
Lecture Time:	Tue, Thu, 12:00PM - 1:20PM
Lecture Location:	CCSB 1.0202
Instructor:	Dr. Christoph Lauter cqlauter@utep.edu 915-747-5939
Instructors Office Hours and Location:	Email anytime, Phone during business hours, Office Mon-Thur 10:30AM-11:45AM
Class Web page:	https://www.christoph-lauter.org/utep-numerical/
Prerequisites:	None. Please bring motivation and joy to class.
Description:	<p>While the first computers in the history of computing were fully dedicated to numerical applications, computer science nowadays mainly focuses on textual information storage and handling. However, in certain fields, the actual numerical compute power of modern machines is still needed and gets leveraged in novel applications such as machine and deep learning.</p> <p>This course gives an overview of the hardware and, in particular, software techniques used for applied numerical computing: how long integers can be stored and manipulated inside a computer, e.g. to allow for strong asymmetric encryption, how real numbers can be represented and manipulate, for example in graphical processing, how signals can be represented and manipulated in software systems, how basic and advanced linear algebra software packages can be designed and then utilized to solve real-life problems such as compressed sensing, dictionary learning or medical imaging, MP3 and JPEG data compression, ADSL transmission, 5G mobile service etc.</p>
Student Learning Outcomes:	See appendix below. Student progress on the learning outcomes will be assessed primarily via programming projects (homework) and exams.
Textbooks:	<p>This is a research-oriented special topics course. There is no specific textbook. Students may want to read the following books and articles:</p> <ul style="list-style-type: none">• Muller et al., <i>Handbook of Floating-Point Arithmetic</i>, Springer• Goldberg, <i>What every Computer Scientist should know about Floating-Point Arithmetic</i>

- Overton, *Numerical Computing with IEEE Floating-Point Arithmetic*, SIAM
- Ercegovic et al., *Digital Arithmetic*, Morgan Kaufman
- Oppenheim et al., *Discrete-Time Signal Processing*

Homework assignments: **Assign-** There will be 3 homework assignments. The homework assignments will be programming based. Students can choose the programming language of their liking¹

Workload & Attendance: **Atten-** This course requires participation during the in-class lectures and engagement into the homework assignments. Please form study groups and you are encouraged to discuss the approach and understand the problem. The write up, programming, and actual solutions to the homeworks are team-work but must be the individual work of each student. If you use someone's work for your own, you are committing plagiarism.

Readings: None. See above.

Computers: Students need access to a computer that allows them using a native Linux environment or that allows them `ssh` access to a Linux server, provided by the instructor. For homework assignments that compile in one environment and not the other, the result obtained on the system provided by the instructor will be deemed the canonical result.

Tests and Testing Policy: Three tests will be given: two mid-term tests and a final. All tests are cumulative, with an emphasis on recent material. No make up tests are given but for documented medical emergencies.

Grading: Letter grade:

Points range	Grade
[90; 100]	A
[80; 90)	B
[70; 80)	C
[60; 70)	D
[0; 60)	F

Grading breakdown:

Homework	60%
Midterm I	7%
Midterm II	8%
Final exam	20%
Attendance	5%

Students taking this course as CS5390 must answer all extra questions marked in all homework assignments. These extra questions will require students read certain scientific papers in order to answer the homework questions.

¹Reasonable languages only; examples of reasonable languages include C, Python, C++, Java, Fortran.

Make-Ups:

There will be no make-up for missed exams or homework deadlines, unless due to a documented medical or family emergency. Providing the right kind of documentation for emergencies is the student's responsibility.

Approximate Schedule:**Course**

The course will be spread out **approximately** according the following schedule:

Week 1	Representation of Unsigned & Signed Integers
Week 2	Addition of Integers, Multiplication of Integers
Week 3	Karatsuba, Toom-Cook, Cryptography
Week 4	Introduction to IEEE754 Floating-Point Numbers
Week 5	Issues with IEEE754 Floating-Point Arithmetic
Week 6	Midterm I
Week 7	Introduction to Error Analysis
Week 9	Introduction to Linear Algebra
Week 8	Applications of Linear Algebra, Interpolation, Dictionary Learning
Week 9	Introduction to Signal Processing
Week 10	Midterm II
Week 11	Fast Fourier Transform, ADSL, MP3, JPEG
Week 12	Goertzel Algorithm, FSK Modulation, DTMF
Week 13	Signal Processing Applications, Medical Imaging
Week 14	Revision, Wrap-Up

This course schedule is **approximative**. In the case the instructor thinks that students need more time to better understand a subject (or less time because they are really, really smart), the instructor will deviate from the schedule. This may affect the dates the midterm exams are given.

Academic Honesty:

- Students are expected to conduct themselves in a professional and courteous manner, as prescribed by the Standards of Conduct: <https://www.utep.edu/hoop/section-2/studentconduct-and-discipline.html>
- Submitted work should be unmistakably your own. You may not transcribe or copy a solution taken from another person, book, or other source (e.g., a web page). Professors are required to report academic dishonesty and any other violation of the Standards of Conduct to the Dean of Students.

- Permitted collaboration: Students may discuss requirements, background information, test sets, solution strategies, and the output of their programs. However, implementations and documentation must be their own creative work. Students are required to document advice received from others and all resources utilized in the preparation of their assignments.
- If academic dishonesty is suspected: The Dean of Students office will be contacted for adjudication. A temporary “incomplete” grade will be issued if their investigation extends beyond the grading period.

Disabilities & Accommodations:

If you have a disability and need accommodations, please contact The Center for Accommodations and Support Services (CASS) at 915-747-5148, or by email to cass@utep.edu. For additional information, please visit the CASS website at www.sa.utep.edu/cass.

COVID19 statement:

- Please stay home if you have been diagnosed with COVID-19 or are experiencing COVID-19 symptoms. If you are feeling unwell, please let me know as soon as possible, so that we can work on appropriate accommodations. If you have tested positive for COVID-19, you are encouraged to report your results to covidaction@utep.edu, so that the Dean of Students Office can provide you with support and help with communication with your professors.
- The Center for Disease Control and Prevention recommends that people in areas of substantial or high COVID-19 transmission wear face masks when indoors in groups of people. The best way that Miners can take care of Miners is to get the vaccine. If you still need the vaccine, it is widely available in the El Paso area. For more information about the current infection rates, testing, and vaccinations, please visit epstrong.org.

Cell phones:

Please silence your mobile devices or put them into a vibrate mode for the duration of class - they are disruptive for your fellow students. No mobile devices (cell phones, PDAs, laptops etc.) are allowed during the exams and will result in your expulsion from the test.

Appendix: CS4390 & CS5390 Learning Outcomes

Level 1:

- Explain positional representation of integers.
- Explain negating integers in various representation of signed integers.
- Analyze the complexity of addition algorithms for integers, understand the optimality of traditional addition algorithms.
- Analyze the complexity of schoolbook multiplication, Karatsuba multiplication and Toom-Cook multiplication.
- Understand the requirements for arbitrary length integers in asymmetric cryptography.
- Encode and decode real numbers as IEEE754 Floating-Point Numbers in various formats.
- Understand basic Wilkinson and Higham Error Analysis.
- Understand the need for Linear Algebra Packages in High Performance Computing.
- Explain the use of Linear Algebra for Interpolation, Dictionary Learning and other Multimedia Applications.
- Explain the principles of Digital Signal Processing with Time and Value-Discrete Signals.
- Explain the principles of MP3 audio compression.
- Explain the principles of digital medical imaging, such as MRT.

Level 2:

- Explain positional representation of integers when machine integers are used for radices $\beta = 2^{32}$ or $\beta = 2^{32}$.
- Explain negation of integers in two's complement when machine integers are used for radices $\beta = 2^{32}$ or $\beta = 2^{32}$.
- Design algorithms for addition of integers with machine integers used for radices $\beta = 2^{32}$ or $\beta = 2^{32}$ when no carry generating machine operation is available.
- Analyze the complexity of multiplication through Fast Fourier Transforms in a finite field.
- Understand the effects of rounding on IEEE754 Arithmetic.
- Understand advanced IEEE754 Floating-Point Algorithms like Fast2Sum and Fast2Mul with FMA.
- Explain the function of Linear Algebra Routines like Matrix-Matrix Multiplication or Gaussian Elimination.

- Explain Fast Fourier Transform on Time and Value-Discrete Signals.
- Explain the principles of FSK modulation and demodulation.

Level 3:

- Implement efficient routines for arbitrary length unsigned integers.
- Implement negation of arbitrary length signed integers in two's complement.
- Implement optimal addition of arbitrary length integers.
- Implement schoolbook, Karatsuba and Toom-Cook multiplication for arbitrary length integers.
- Implement routines to decode IEEE754 Floating-Point and encode them, with rounding in various rounding directions.
- Implement basic Linear Algebra routines including Gaussian Elimination/ LU decomposition.
- Implement Fast Fourier Transform.
- Implement basic signal processing routines such as sinusoid generation or frequency detection with the Goertzel algorithm.
- Implement FSK modulation and demodulation.