

COMPUT & MATH ENGR (CME)

CME 100. Vector Calculus for Engineers. 5 Units.

Computation and visualization using MATLAB. Differential vector calculus: vector-valued functions, analytic geometry in space, functions of several variables, partial derivatives, gradient, linearization, unconstrained maxima and minima, Lagrange multipliers and applications to trajectory simulation, least squares, and numerical optimization. Introduction to linear algebra: matrix operations, systems of algebraic equations with applications to coordinate transformations and equilibrium problems. Integral vector calculus: multiple integrals in Cartesian, cylindrical, and spherical coordinates, line integrals, scalar potential, surface integrals, Green's, divergence, and Stokes' theorems. Numerous examples and applications drawn from classical mechanics, fluid dynamics and electromagnetism. Prerequisites: knowledge of single-variable calculus equivalent to the content of Math 19-21 (e.g., 5 on Calc BC, 4 on Calc BC with Math 21, 5 on Calc AB with Math 21). Placement diagnostic (recommendation non-binding) at: <https://exploreddegrees.stanford.edu/undergraduatedegreesandprograms/#aptext>.

Same as: ENGR 154

CME 100A. Vector Calculus for Engineers, ACE. 6 Units.

Students attend CME100/ENGR154 lectures with additional recitation sessions; two to four hours per week, emphasizing engineering mathematical applications and collaboration methods. Enrollment by department permission only. Prerequisite: must be enrolled in the regular CME100-01 or 02. Application at: <https://engineering.stanford.edu/students/programs/engineering-diversity-programs/additional-calculus-engineers>.

CME 102. Ordinary Differential Equations for Engineers. 5 Units.

Analytical and numerical methods for solving ordinary differential equations arising in engineering applications are presented. For analytical methods students learn to solve linear and non-linear first order ODEs; linear second order ODEs; and Laplace transforms. Numerical methods using MATLAB programming tool kit are also introduced to solve various types of ODEs including: first and second order ODEs, higher order ODEs, systems of ODEs, initial and boundary value problems, finite differences, and multi-step methods. This also includes accuracy and linear stability analyses of various numerical algorithms which are essential tools for the modern engineer. This class is foundational for professional careers in engineering and as a preparation for more advanced classes at the undergraduate and graduate levels. Prerequisites: knowledge of single-variable calculus equivalent to the content of Math 19-21 (e.g., 5 on Calc BC, 4 on Calc BC with Math 21, 5 on Calc AB with Math 21). Placement diagnostic (recommendation non-binding) at: <https://exploreddegrees.stanford.edu/undergraduatedegreesandprograms/#aptext>.

Same as: ENGR 155A

CME 102A. Ordinary Differential Equations for Engineers, ACE. 6 Units.

Students attend CME102/ENGR155A lectures with additional recitation sessions; two to four hours per week, emphasizing engineering mathematical applications and collaboration methods. Prerequisite: students must be enrolled in the regular section (CME102) prior to submitting application at: <https://engineering.stanford.edu/students/programs/engineering-diversity-programs/additional-calculus-engineers>.

CME 104. Linear Algebra and Partial Differential Equations for Engineers. 5 Units.

Linear algebra: systems of algebraic equations, Gaussian elimination, undetermined and overdetermined systems, coupled systems of ordinary differential equations, LU factorization, eigensystem analysis, normal modes. Linear independence, vector spaces, subspaces and basis. Numerical analysis applied to structural equilibrium problems, electrical networks, and dynamic systems. Fourier series with applications, partial differential equations arising in science and engineering, analytical solutions of partial differential equations. Applications in heat and mass transport, mechanical vibration and acoustic waves, transmission lines, and fluid mechanics. Numerical methods for solution of partial differential equations: iterative techniques, stability and convergence, time advancement, implicit methods, von Neumann stability analysis. Examples and applications drawn from a variety of engineering fields. Prerequisite: CME102/ENGR155A.

Same as: ENGR 155B

CME 104A. Linear Algebra and Partial Differential Equations for Engineers, ACE. 6 Units.

Students attend CME104/ENGR155B lectures with additional recitation sessions; two to four hours per week, emphasizing engineering mathematical applications and collaboration methods. Prerequisite: students must be enrolled in the regular section (CME104) prior to submitting application at: <https://engineering.stanford.edu/students/programs/engineering-diversity-programs/additional-calculus-engineers>.

CME 106. Introduction to Probability and Statistics for Engineers. 4 Units.

Probability: random variables, independence, and conditional probability; discrete and continuous distributions, moments, distributions of several random variables. Numerical simulation using Monte Carlo techniques. Topics in mathematical statistics: random sampling, point estimation, confidence intervals, hypothesis testing, non-parametric tests, regression and correlation analyses. Numerous applications in engineering, manufacturing, reliability and quality assurance, medicine, biology, and other fields. Prerequisite: CME100/ENGR154 or Math 51 or 52.

Same as: ENGR 155C

CME 107. Introduction to Machine Learning. 3-5 Units.

Introduction to machine learning. Formulation of supervised and unsupervised learning problems. Regression and classification. Data standardization and feature engineering. Loss function selection and its effect on learning. Regularization and its role in controlling complexity. Validation and overfitting. Robustness to outliers. Simple numerical implementation. Experiments on data from a wide variety of engineering and other disciplines. Undergraduate students should enroll for 5 units, and graduate students should enroll for 3 units. Prerequisites: ENGR 108; EE 178 or CS 109; CS106A or equivalent.

Same as: EE 104

CME 108. Introduction to Scientific Computing. 3 Units.

Introduction to Scientific Computing Numerical computation for mathematical, computational, physical sciences and engineering: error analysis, floating-point arithmetic, nonlinear equations, numerical solution of systems of algebraic equations, banded matrices, least squares, unconstrained optimization, polynomial interpolation, numerical differentiation and integration, numerical solution of ordinary differential equations, truncation error, numerical stability for time dependent problems and stiffness. Implementation of numerical methods in MATLAB programming assignments. Prerequisites: MATH 51, 52, 53; prior programming experience (MATLAB or other language at level of CS 106A or higher).

Same as: MATH 114

CME 10A. Explorations in Calculus. 1 Unit.

In this course, we will explore the big ideas of calculus, through open, visual, and creative mathematics tasks. Students will be invited to think about what calculus is all about and why it matters. This course will benefit all students ζ whether or not you have taken a calculus class. Students will work collaboratively in problem solving through a supportive community of mathematics learners. This course has three goals ζ to give you a different mathematics experience that could reshape your relationship with mathematics, to provide you with a basis for success in future courses at Stanford, and to teach you the important ideas that pervade calculus. As a community, we will cultivate the positive ideas and mindsets that shape productive learning.

CME 151A. Interactive Data Visualization in D3. 1 Unit.

This four-week short course introduces D3, a powerful tool for creating interactive data visualizations on the web (d3js.org). The class is geared toward scientists and engineers who want to better communicate their personal projects and research through visualizations on the web. The class will cover the basics of D3: inputting data, creating scales and axes, and adding transitions and interactivity, as well as some of the most used libraries: stack, cluster and force layouts. The class will be based on short workshops and a final project. A background in programming methodology at the level of CS106A is assumed. The course will make use of Javascript, experience is recommended but not necessary.

CME 187. Mathematical Population Biology. 3 Units.

Mathematical models in population biology, in biological areas including demography, ecology, epidemiology, evolution, and genetics. Mathematical approaches include techniques in areas such as combinatorics, differential equations, dynamical systems, linear algebra, probability, and stochastic processes. Math 50 or 60 series is required, and at least two of (Bio 81, Bio 82, Bio 85) are strongly recommended. Same as: BIO 187

CME 192. Introduction to MATLAB. 1 Unit.

This short course runs for the first four weeks/eight lectures of the quarter and is offered each quarter during the academic year. It is highly recommended for students with no prior programming experience who are expected to use MATLAB in math, science, or engineering courses. It will consist of interactive lectures and application-based assignments. The goal of the short course is to make students fluent in MATLAB and to provide familiarity with its wide array of features. The course covers an introduction of basic programming concepts, data structures, and control/flow; and an introduction to scientific computing in MATLAB, scripts, functions, visualization, simulation, efficient algorithm implementation, toolboxes, and more.

CME 193. Introduction to Scientific Python. 1 Unit.

This short course runs for the first four weeks of the quarter. It is recommended for students who are familiar with programming at least at the level of CS106A and want to translate their programming knowledge to Python with the goal of becoming proficient in the scientific computing and data science stack. Lectures will be interactive with a focus on real world applications of scientific computing. Technologies covered include Numpy, SciPy, Pandas, Scikit-learn, and others. Topics will be chosen from Linear Algebra, Optimization, Machine Learning, and Data Science. Prior knowledge of programming will be assumed, and some familiarity with Python is helpful, but not mandatory.

CME 195. Introduction to R. 1 Unit.

This short course runs for four weeks and is offered in fall and spring. It is recommended for students who want to use R in statistics, science or engineering courses, and for students who want to learn the basics of data science with R. The goal of the short course is to familiarize students with some of the most important R tools for data analysis. Lectures will focus on learning by example and assignments will be application-driven. No prior programming experience is assumed. Same as: STATS 195

CME 197. Human-Centered Design Methods in Data Science. 1 Unit.

In today's society, the most pressing data science problems we face exist in a complex sociotechnical ecosystem and cannot be solved using the numbers alone. In this five-week short course, students will learn how to apply human-centered design methods to solve data science problems and how to pair traditional data with a diversity of other types of data to redefine problems and gain innovative insight. The course will focus on empathy-based frameworks to analyze data, problem definition and redefinition, and ideation. Additional skills in critique and storytelling will also be covered. Classes will be highly interactive and team-based. This course will offer skills in support of the teams working toward the Big Earth Hackathon Wildland Fire challenge (CEE 265H, EARTH 165H, EARTH 265H).

Same as: EARTH 197

CME 200. Linear Algebra with Application to Engineering Computations. 3 Units.

Computer based solution of systems of algebraic equations obtained from engineering problems and eigen-system analysis, Gaussian elimination, effect of round-off error, operation counts, banded matrices arising from discretization of differential equations, ill-conditioned matrices, matrix theory, least square solution of unsolvable systems, solution of non-linear algebraic equations, eigenvalues and eigenvectors, similar matrices, unitary and Hermitian matrices, positive definiteness, Cayley-Hamilton theory and function of a matrix and iterative methods. Prerequisite: familiarity with computer programming, and MATH51. Same as: ME 300A

CME 204. Partial Differential Equations in Engineering. 3 Units.

Geometric interpretation of partial differential equation (PDE) characteristics; solution of first order PDEs and classification of second-order PDEs; self-similarity; separation of variables as applied to parabolic, hyperbolic, and elliptic PDEs; special functions; eigenfunction expansions; the method of characteristics. If time permits, Fourier integrals and transforms, Laplace transforms. Prerequisite: CME 200/ME 300A, equivalent, or consent of instructor. Same as: ME 300B

CME 206. Introduction to Numerical Methods for Engineering. 3 Units.

Numerical methods from a user's point of view. Lagrange interpolation, splines. Integration: trapezoid, Romberg, Gauss, adaptive quadrature; numerical solution of ordinary differential equations: explicit and implicit methods, multistep methods, Runge-Kutta and predictor-corrector methods, boundary value problems, eigenvalue problems; systems of differential equations, stiffness. Emphasis is on analysis of numerical methods for accuracy, stability, and convergence. Introduction to numerical solutions of partial differential equations; Von Neumann stability analysis; alternating direction implicit methods and nonlinear equations. Prerequisites: CME 200/ME 300A, CME 204/ME 300B. Same as: ME 300C

CME 209. Mathematical Modeling of Biological Systems. 3 Units.

The course covers mathematical and computational techniques needed to solve advanced problems encountered in applied bioengineering. Fundamental concepts are presented in the context of their application to biological and physiological problems including cancer, cardiovascular disease, infectious disease, and systems biology. Topics include Taylor's Series expansions, parameter estimation, regression, nonlinear equations, linear systems, optimization, numerical differentiation and integration, stochastic methods, ordinary differential equations and Fourier series. Python, Matlab and other software will be used for weekly assignments and projects. Prerequisites: Math 51, 52, 53; prior programming experience (Matlab or other language at level of CS 106a or higher). Same as: BIOE 209

CME 211. Software Development for Scientists and Engineers. 3 Units.

Basic usage of the Python and C/C++ programming languages are introduced and used to solve representative computational problems from various science and engineering disciplines. Software design principles including time and space complexity analysis, data structures, object-oriented design, decomposition, encapsulation, and modularity are emphasized. Usage of campus wide Linux compute resources: login, file system navigation, editing files, compiling and linking, file transfer, etc. Versioning and revision control, software build utilities, and the LaTeX typesetting software are introduced and used to help complete programming assignments. Prerequisite: introductory programming course equivalent to CS 106A or instructor consent.

CME 212. Advanced Software Development for Scientists and Engineers. 3 Units.

Advanced topics in software development, debugging, and performance optimization are covered. The capabilities and usage of common libraries and frameworks such as BLAS, LAPACK, FFT, PETSc, and MKL/ACML are reviewed. Computer representation of integer and floating point numbers, and interoperability between C/C++ and Fortran is described. More advanced software engineering topics including: representing data in files, signals, unit and regression testing, and build automation. The use of debugging tools including static analysis, gdb, and Valgrind are introduced. An introduction to computer architecture covering processors, memory hierarchy, storage, and networking provides a foundation for understanding software performance. Profiles generated using gprof and perf are used to help guide the performance optimization process. Computational problems from various science and engineering disciplines will be used in assignments. Prerequisites: CME 200 / ME 300A and CME 211.

CME 213. Introduction to parallel computing using MPI, openMP, and CUDA. 3 Units.

This class will give hands-on experience with programming multicore processors, graphics processing units (GPU), and parallel computers. The focus will be on the message passing interface (MPI, parallel clusters) and the compute unified device architecture (CUDA, GPU). Topics will include multithreaded programs, GPU computing, computer cluster programming, C++ threads, OpenMP, CUDA, and MPI. Pre-requisites include C++, templates, debugging, UNIX, makefile, numerical algorithms (differential equations, linear algebra).

Same as: ME 339

CME 214. Software Design in Modern Fortran for Scientists and Engineers. 3 Units.

This course introduces software design and development in modern Fortran. Course covers the functional, object-oriented-, and parallel programming features introduced in the Fortran 95, 2003, and 2008 standards, respectively, in the context of numerical approximations to ordinary and partial differential equations; introduces object-oriented design and design schematics based on the Unified Modeling Language (UML) structure, behavior, and interaction diagrams; cover the basic use of several open-source tools for software building, testing, documentation generation, and revision control. Recommended: Familiarity with programming in Fortran 90, basic numerical analysis and linear algebra, or instructor approval.

Same as: EARTH 214

CME 215A. Advanced Computational Fluid Dynamics. 3 Units.

High resolution schemes for capturing shock waves and contact discontinuities; upwinding and artificial diffusion; LED and TVD concepts; alternative flow splittings; numerical shock structure. Discretization of Euler and Navier Stokes equations on unstructured meshes; the relationship between finite volume and finite element methods. Time discretization; explicit and implicit schemes; acceleration of steady state calculations; residual averaging; math grid preconditioning. Automatic design; inverse problems and aerodynamic shape optimization via adjoint methods. Pre- or corequisite: 214B or equivalent.

Same as: AA 215A

CME 215B. Advanced Computational Fluid Dynamics. 3 Units.

High resolution schemes for capturing shock waves and contact discontinuities; upwinding and artificial diffusion; LED and TVD concepts; alternative flow splittings; numerical shock structure. Discretization of Euler and Navier Stokes equations on unstructured meshes; the relationship between finite volume and finite element methods. Time discretization; explicit and implicit schemes; acceleration of steady state calculations; residual averaging; math grid preconditioning. Automatic design; inverse problems and aerodynamic shape optimization via adjoint methods. Pre- or corequisite: 214B or equivalent.

Same as: AA 215B

CME 216. Machine Learning for Computational Engineering.. 3 Units.

Linear and kernel support vector machines, deep learning, deep neural networks, generative adversarial networks, physics-based machine learning, forward and reverse mode automatic differentiation, optimization algorithms for machine learning, TensorFlow, PyTorch.

Same as: ME 343

CME 217. Analytics Accelerator. 1-3 Unit.

This is a multidisciplinary graduate level course designed to give students hands-on experience working in teams through real-world project-based research and experiential classroom activities. Students work in dynamic teams with the support of course faculty and mentors, researching preselected topics focused on COVID-19 during fall 2020 with the option to continue into winter 2021. Students apply a computational and data analytics lens and will use design thinking methodology. The course exposes students to ethics, emotional intelligence, unintended consequences of their work and team building supported by relevant lectures on data science and med/bio topics. Pre-requisites: none. The course application generally opens 5-6 weeks before registration for each quarter. If you missed the application for the quarter, please submit your application anyway to be added to the waitlist and to receive information regarding upcoming quarters. <https://forms.gle/oLtUe7dMKGy8bb2Z9>.

Same as: BIODS 217

CME 232. Introduction to Computational Mechanics. 3 Units.

Provides an introductory overview of modern computational methods for problems arising primarily in mechanics of solids and is intended for students from various engineering disciplines. The course reviews the basic theory of linear solid mechanics and introduces students to the important concept of variational forms, including the principle of minimum potential energy and the principles of virtual work. Specific model problems that will be considered include deformation of bars, beams and membranes, plates, and problems in plane elasticity (plane stress, plane strain, axisymmetric elasticity). The variational forms of these problems are used as the starting point for developing the finite element method (FEM) and boundary element method (BEM) approaches providing an important connection between mechanics and computational methods.

Same as: ME 332

CME 241. Reinforcement Learning for Stochastic Control Problems in Finance. 3 Units.

This course will explore a few problems in Mathematical Finance through the lens of Stochastic Control, such as Portfolio Management, Derivatives Pricing/Hedging and Order Execution. For each of these problems, we formulate a suitable Markov Decision Process (MDP), develop Dynamic Programming (DP) solutions, and explore Reinforcement Learning (RL) algorithms. The course emphasizes the theory of DP/RL as well as modeling the practical nuances of these finance problems, and strengthening the understanding through plenty of coding exercises of the methods. No pre-requisite coursework expected, but a foundation in undergraduate Probability, basic familiarity with Finance, and Python coding skills are required. Dynamic Programming or Reinforcement Learning background not required.

Same as: MS&E 346

CME 243. Risk Analytics and Management in Finance and Insurance. 2-4 Units.

Market risk and credit risk, credit markets. Back testing, stress testing and Monte Carlo methods. Logistic regression, generalized linear models and generalized mixed models. Loan prepayment and default as competing risks. Survival and hazard functions, correlated default intensities, frailty and contagion. Risk surveillance, early warning and adaptive control methodologies. Banking and bank regulation, asset and liability management. Prerequisite: STATS 240 or equivalent. Same as: STATS 243

CME 245. Topics in Mathematical and Computational Finance. 1 Unit.

Description: Current topics for enrolled students in the MCF program: This course is an introduction to computational, statistical, and optimizations methods and their application to financial markets. Class will consist of lectures and real-time problem solving. Topics: Python & R programming, interest rates, Black-Scholes model, financial time series, capital asset pricing model (CAPM), options, optimization methods, and machine learning algorithms. Appropriate for anyone with a technical and solid applied math background interested in honing skills in quantitative finance. Prerequisite: basic statistics and exposure to programming. Can be repeated up to three times.

CME 249. Using Design for Effective Data Analysis. 1 Unit.

Teams of students use techniques in applied and computational mathematics to tackle problems with real world data sets. Application of design methodology adapted for data analysis will be emphasized; leverage design thinking to come up with efficient and effective data driven insights; explore design thinking methodology in small group setting; apply design thinking to a specific data centric problem and make professional group presentation of the results. Limited enrollment. Prerequisites: CME100/102/104 or equivalents, or instructor consent. Recommended: CME106/108 and familiarity with programming at the level of CME 192/193.

CME 250. Introduction to Machine Learning. 1 Unit.

A Short course presenting the principles behind when, why, and how to apply modern machine learning algorithms. We will discuss a framework for reasoning about when to apply various machine learning techniques, emphasizing questions of over-fitting/under-fitting, regularization, interpretability, supervised/unsupervised methods, and handling of missing data. The principles behind various algorithms—the why and how of using them—will be discussed, while some mathematical detail underlying the algorithms—including proofs—will not be discussed. Unsupervised machine learning algorithms presented will include k-means clustering, principal component analysis (PCA), and independent component analysis (ICA). Supervised machine learning algorithms presented will include support vector machines (SVM), classification and regression trees (CART), boosting, bagging, and random forests. Imputation, the lasso, and cross-validation concepts will also be covered. The R programming language will be used for examples, though students need not have prior exposure to R. Prerequisite: undergraduate-level linear algebra and statistics; basic programming experience (R/Matlab/Python).

CME 250A. Machine Learning on Big Data. 1 Unit.

A short course presenting the application of machine learning methods to large datasets. Topics include: brief review of the common issues of machine learning, such as, memorizing/overfitting vs learning, test/train splits, feature engineering, domain knowledge, fast/simple/dumb learners vs slow/complex/smart learners; moving your model from your laptop into a production environment using Python (scikit) or R on small data (laptop sized) at first; building math clusters using the open source H2O product to tackle Big Data, and finally to some model building on terabyte sized datasets. Prerequisites: basic knowledge of statistics, matrix algebra, and unix-like operating systems; basic file and text manipulation skills with unix tools: pipes, cut, paste, grep, awk, sed, sort, zip; programming skill at the level of CME211 or CS106A.

CME 250Q. Introduction to Quantum Computing and Quantum Algorithms. 1 Unit.

This course will cover the basic formalism of quantum states and quantum measurements, and introduce the circuit model of quantum computation. Basic results such as the Solovay-Kitaev theorem, no-cloning theorem, quantum entanglement and Bell's inequality will be discussed followed by the quantum Fourier transform (QFT) and quantum phase estimation (QPE), and cover some of its important applications such as the celebrated Shor's algorithm for integer factorization (other applications will be mentioned but not discussed in detail), Grover's algorithm for quantum search is covered next, and lower bounds for query complexity in this context; some basic concepts of quantum error correction and quantum entropy, distance between quantum states, subadditivity and strong subadditivity of von Neumann quantum entropy will also be covered. Time permitting, we will discuss some advanced algorithms such as the HHL algorithm for matrix inversion, VQE (variational quantum eigensolver) and the QAOA algorithm for optimization. Requires programming in Python, where the goal will be to familiarize the students to available software for quantum algorithm development, existing libraries, and also run some simple programs on a real quantum computer. Prerequisites: Linear algebra at the level of CME 200 / MATH 104, basic knowledge of group theory, and programming in Python. Additionally, some knowledge of real analysis will be helpful.

CME 251. Geometric and Topological Data Analysis. 3 Units.

Mathematical and computational tools for the analysis of data with geometric content, such images, videos, 3D scans, GPS traces – as well as for other data embedded into geometric spaces. Global and local geometry descriptors allowing for various kinds of invariances. The rudiments of computational topology and persistent homology on sampled spaces. Clustering and other unsupervised techniques. Spectral methods for graph data. Linear and non-linear dimensionality reduction techniques. Alignment, matching, and map computation between geometric data sets. Function spaces and functional maps. Networks of data sets and joint analysis for segmentation and labeling. Deep learning on irregular geometric data. Prerequisites: discrete algorithms at the level of CS161; linear algebra at the level of Math51 or CME103. Same as: CS 233

CME 253. Introduction to GPU Computing and CUDA. 1 Unit.

Covers the fundamentals of accelerating applications with GPUs (Graphics Processing Units); GPU programming with CUDA and OpenACC, debugging, thrust/CUB, profiling, optimization, debugging, and other CUDA tools. Libraries to easily accelerate compute code will be presented and deployment on larger systems will be addressed, including multi-GPU environments. Several practical examples will be detailed, including deep learning. Pre-requisite: knowledge of C/C++ at the level of CME211 or CS106b.

CME 257. Advanced Topics in Scientific Computing with Julia. 1 Unit.

This course will rapidly introduce students to the Julia programming language, with the goal of giving students the knowledge and experience necessary to navigate the language and package ecosystem while using Julia for their own scientific computing needs. The course will begin with learning the basics of Julia, and then introduce students to git version control and package development. Additional topics include: common packages, parallelism, interfacing with shared object libraries, and aspects of Julia's implementation (e.g. core numerical linear algebra). Lectures will be interactive, with an emphasis on collaboration and learning by example. Prerequisites: Data structures at the level of CS106B, experience with one or more scientific computing languages (e.g. Python, Matlab, or R), and some familiarity with the Unix shell. No prior experience with Julia or git is required.

CME 262. Imaging with Incomplete Information. 3-4 Units.

Statistical and computational methods for inferring images from incomplete data. Bayesian inference methods are used to combine data and quantify uncertainty in the estimate. Fast linear algebra tools are used to solve problems with many pixels and many observations. Applications from several fields but mainly in earth sciences. Prerequisites: Linear algebra and probability theory. Same as: CEE 362G

CME 263. Introduction to Linear Dynamical Systems. 3 Units.

Applied linear algebra and linear dynamical systems with applications to circuits, signal processing, communications, and control systems. Topics: least-squares approximations of over-determined equations, and least-norm solutions of underdetermined equations. Symmetric matrices, matrix norm, and singular-value decomposition. Eigenvalues, left and right eigenvectors, with dynamical interpretation. Matrix exponential, stability, and asymptotic behavior. Multi-input/multi-output systems, impulse and step matrices; convolution and transfer-matrix descriptions. Control, reachability, and state transfer; observability and least-squares state estimation. Prerequisites: Linear algebra and matrices as in ENGR 108 or MATH 104; ordinary differential equations and Laplace transforms as in EE 102B or CME 102. Same as: EE 263

CME 279. Computational Biology: Structure and Organization of Biomolecules and Cells. 3 Units.

Computational techniques for investigating and designing the three-dimensional structure and dynamics of biomolecules and cells. These computational methods play an increasingly important role in drug discovery, medicine, bioengineering, and molecular biology. Course topics include protein structure prediction, protein design, drug screening, molecular simulation, cellular-level simulation, image analysis for microscopy, and methods for solving structures from crystallography and electron microscopy data. Prerequisites: elementary programming background (CS 106A or equivalent) and an introductory course in biology or biochemistry. Same as: BIOE 279, BIOMEDIN 279, BIOPHYS 279, CS 279

CME 285. Computational Modeling in the Cardiovascular System. 3 Units.

This course introduces computational modeling methods for cardiovascular blood flow and physiology. Topics in this course include analytical and computational methods for solutions of flow in deformable vessels, one-dimensional equations of blood flow, cardiovascular anatomy, lumped parameter models, vascular trees, scaling laws, biomechanics of the circulatory system, and 3D patient specific modeling with finite elements; course will provide an overview of the diagnosis and treatment of adult and congenital cardiovascular diseases and review recent research in the literature in a journal club format. Students will use SimVascular software to do clinically-oriented projects in patient specific blood flow simulations. Pre-requisites: CME102, ME133 and CME192. Same as: BIOE 285, ME 285

CME 291. Master's Research. 1-6 Unit.

Students require faculty sponsor. (Staff).

CME 292. Advanced MATLAB for Scientific Computing. 1 Unit.

Short course running first four weeks of the quarter (8 lectures) with interactive online lectures and application based assignment. Students will access the lectures and assignments on <https://suclass.stanford.edu>. Students will be introduced to advanced MATLAB features, syntaxes, and toolboxes not traditionally found in introductory courses. Material will be reinforced with in-class examples, demos, and homework assignment involving topics from scientific computing. MATLAB topics will be drawn from: advanced graphics (2D/3D plotting, graphics handles, publication quality graphics, animation), MATLAB tools (debugger, profiler), code optimization (vectorization, memory management), object-oriented programming, compiled MATLAB (MEX files and MATLAB coder), interfacing with external programs, toolboxes (optimization, parallel computing, symbolic math, PDEs). Scientific computing topics will include: numerical linear algebra, numerical optimization, ODEs, and PDEs.

CME 298. Basic Probability and Stochastic Processes with Engineering Applications. 3 Units.

Calculus of random variables and their distributions with applications. Review of limit theorems of probability and their application to statistical estimation and basic Monte Carlo methods. Introduction to Markov chains, random walks, Brownian motion and basic stochastic differential equations with emphasis on applications from economics, physics and engineering, such as filtering and control. Prerequisites: exposure to basic probability. Same as: MATH 158

CME 300. First Year Seminar Series. 1 Unit.

Required for first-year ICME Ph.D. students; recommended for first-year ICME M.S. students. Presentations about research at Stanford by faculty and researchers from Engineering, H&S, and organizations external to Stanford. May be repeated for credit.

CME 302. Numerical Linear Algebra. 3 Units.

Solution of linear systems, accuracy, stability, LU, Cholesky, QR, least squares problems, singular value decomposition, eigenvalue computation, iterative methods, Krylov subspace, Lanczos and Arnoldi processes, conjugate gradient, GMRES, direct methods for sparse matrices. Prerequisites: CME 108, MATH 114, MATH 104.

CME 303. Partial Differential Equations of Applied Mathematics. 3 Units.

First-order partial differential equations; method of characteristics; weak solutions; elliptic, parabolic, and hyperbolic equations; Fourier transform; Fourier series; and eigenvalue problems. Prerequisite: Basic coursework in multivariable calculus and ordinary differential equations, and some prior experience with a proof-based treatment of the material as in Math 171 or Math 61CM.nnNOTE: Undergraduates require instructor permission to enroll. Undergraduates interested in taking the course should contact the instructor for permission, providing information about relevant background such as performance in prior coursework, reading, etc. Same as: MATH 220

CME 305. Discrete Mathematics and Algorithms. 3 Units.

Topics: Basic Algebraic Graph Theory, Matroids and Minimum Spanning Trees, Submodularity and Maximum Flow, NP-Hardness, Approximation Algorithms, Randomized Algorithms, The Probabilistic Method, and Spectral Sparsification using Effective Resistances. Topics will be illustrated with applications from Distributed Computing, Machine Learning, and large-scale Optimization. Prerequisites: CS 261 is highly recommended, although not required. Same as: MS&E 316

CME 306. Numerical Solution of Partial Differential Equations. 3 Units.

Hyperbolic partial differential equations: stability, convergence and qualitative properties; nonlinear hyperbolic equations and systems; combined solution methods from elliptic, parabolic, and hyperbolic problems. Examples include: Burger's equation, Euler equations for compressible flow, Navier-Stokes equations for incompressible flow. Prerequisites: MATH 220 or CME 302.nnNOTE: Undergraduates require instructor permission to enroll. Undergraduates interested in taking the course should contact the instructor for permission, providing information about relevant background such as performance in prior coursework, reading, etc. Same as: MATH 226

CME 307. Optimization. 3 Units.

Applications, theories, and algorithms for finite-dimensional linear and nonlinear optimization problems with continuous variables. Elements of convex analysis, first- and second-order optimality conditions, sensitivity and duality. Algorithms for unconstrained optimization, and linearly and nonlinearly constrained problems. Modern applications in communication, game theory, auction, and economics. Prerequisites: MATH 113, 115, or equivalent. Same as: MS&E 311

CME 308. Stochastic Methods in Engineering. 3 Units.

The basic limit theorems of probability theory and their application to maximum likelihood estimation. Basic Monte Carlo methods and importance sampling. Markov chains and processes, random walks, basic ergodic theory and its application to parameter estimation. Discrete time stochastic control and Bayesian filtering. Diffusion approximations, Brownian motion and an introduction to stochastic differential equations. Examples and problems from various applied areas. Prerequisites: exposure to probability and background in analysis. Same as: MATH 228, MS&E 324

CME 309. Randomized Algorithms and Probabilistic Analysis. 3 Units.

Randomness pervades the natural processes around us, from the formation of networks, to genetic recombination, to quantum physics. Randomness is also a powerful tool that can be leveraged to create algorithms and data structures which, in many cases, are more efficient and simpler than their deterministic counterparts. This course covers the key tools of probabilistic analysis, and application of these tools to understand the behaviors of random processes and algorithms. Emphasis is on theoretical foundations, though we will apply this theory broadly, discussing applications in machine learning and data analysis, networking, and systems. Topics include tail bounds, the probabilistic method, Markov chains, and martingales, with applications to analyzing random graphs, metric embeddings, random walks, and a host of powerful and elegant randomized algorithms. Prerequisites: CS 161 and STAT 116, or equivalents and instructor consent. Same as: CS 265

CME 321A. Mathematical Methods of Imaging. 3 Units.

Image denoising and deblurring with optimization and partial differential equations methods. Imaging functionals based on total variation and ℓ_1 minimization. Fast algorithms and their implementation. Same as: MATH 221A

CME 321B. Mathematical Methods of Imaging. 3 Units.

Array imaging using Kirchhoff migration and beamforming, resolution theory for broad and narrow band array imaging in homogeneous media, topics in high-frequency, variable background imaging with velocity estimation, interferometric imaging methods, the role of noise and inhomogeneities, and variational problems that arise in optimizing the performance of array imaging algorithms. Same as: MATH 221B

CME 322. Spectral Methods in Computational Physics. 3 Units.

Data analysis, spectra and correlations, sampling theorem, nonperiodic data, and windowing; spectral methods for numerical solution of partial differential equations; accuracy and computational cost; fast Fourier transform, Galerkin, collocation, and Tau methods; spectral and pseudospectral methods based on Fourier series and eigenfunctions of singular Sturm-Liouville problems; Chebyshev, Legendre, and Laguerre representations; convergence of eigenfunction expansions; discontinuities and Gibbs phenomenon; aliasing errors and control; efficient implementation of spectral methods; spectral methods for complicated domains; time differencing and numerical stability. Same as: ME 408

CME 323. Distributed Algorithms and Optimization. 3 Units.

The emergence of clusters of commodity machines with parallel processing units has brought with it a slew of new algorithms and tools. Many fields such as Machine Learning and Optimization have adapted their algorithms to handle such clusters. Topics include distributed and parallel algorithms for: Optimization, Numerical Linear Algebra, Machine Learning, Graph analysis, Streaming algorithms, and other problems that are challenging to scale on a commodity cluster. The class will focus on analyzing parallel and distributed programs, with some implementation using Apache Spark and TensorFlow. Recommended prerequisites: Discrete math at the level of CS 161 and programming at the level of CS 106A.

CME 328. Advanced Topics in Partial Differential Equations. 3 Units.

Contents change each time and is taught as a topics course, most likely by a faculty member visiting from another institution. May be repeated for credit. Topic in 2012-13: numerical solution of time-dependent partial differential equations is a fundamental tool for modeling and prediction in many areas of science and engineering. In this course we explore the stability, accuracy, efficiency, and appropriateness of specialized temporal integration strategies for different classes of partial differential equations including stiff problems and fully implicit methods, operator splitting and semi-implicit methods, extrapolation methods, multirate time integration, multi-physics problems, symplectic integration, and temporal parallelism. Prerequisites: recommended CME303 and 306 or with instructor's consent.

CME 330. Applied Mathematics in the Chemical and Biological Sciences. 3 Units.

Mathematical solution methods via applied problems including chemical reaction sequences, mass and heat transfer in chemical reactors, quantum mechanics, fluid mechanics of reacting systems, and chromatography. Topics include generalized vector space theory, linear operator theory with eigenvalue methods, phase plane methods, perturbation theory (regular and singular), solution of parabolic and elliptic partial differential equations, and transform methods (Laplace and Fourier). Prerequisites: CME 102/ENGR 155A and CME 104/ENGR 155B, or equivalents. Same as: CHEMENG 300

CME 335. Advanced Topics in Numerical Linear Algebra. 3 Units.

Possible topics: Classical and modern (e.g., focused on provable communication minimization) algorithms for executing dense and sparse-direct factorizations in high-performance, distributed-memory environments; distributed dense eigensolvers, dense and sparse-direct triangular solvers, and sparse matrix-vector multiplication; unified analysis of distributed Interior Point Methods for symmetric cones via algorithms for distributing Jordan algebras over products of second-order cones and Hermitian matrices. May be repeated for credit. Prerequisites: CME 302 and CME 304 (or equivalents).

CME 342. Parallel Methods in Numerical Analysis. 3 Units.

Emphasis is on techniques for obtaining maximum parallelism in numerical algorithms, especially those occurring when solving matrix problems, partial differential equations, and the subsequent mapping onto the computer. Implementation issues on parallel computers. Topics: parallel architecture, programming models (MPI, GPU Computing with CUDA ζ quick review), matrix computations, FFT, fast multiple methods, domain decomposition, graph partitioning, discrete algorithms. Prerequisites: 302 or 200 (ME 300A), 213 or equivalent, or consent of instructor. Recommended: differential equations and knowledge of a high-level programming language such as C or C++ (F90/95 also allowable).

CME 345. Model Reduction. 3 Units.

Model reduction is an indispensable tool for computational-based design and optimization, statistical analysis, embedded computing, and real-time optimal control. It is also essential for scenarios where real-time simulation responses are desired. This course presents the basic mathematical theory for projection-based model reduction. It is intended primarily for graduate students interested in computational sciences and engineering. The course material described below is complemented by a balanced set of theoretical, algorithmic, and Matlab computer programming homework assignments. Prerequisites: Solid foundations in numerical linear algebra (CME 200 or equivalent). Basic numerical methods for ODEs (CME 206 or equivalent). Same as: AA 216

CME 364A. Convex Optimization I. 3 Units.

Convex sets, functions, and optimization problems. The basics of convex analysis and theory of convex programming: optimality conditions, duality theory, theorems of alternative, and applications. Least-squares, linear and quadratic programs, semidefinite programming, and geometric programming. Numerical algorithms for smooth and equality constrained problems; interior-point methods for inequality constrained problems. Applications to signal processing, communications, control, analog and digital circuit design, computational geometry, statistics, machine learning, and mechanical engineering. Prerequisite: linear algebra such as EE263, basic probability.

Same as: EE 364A

CME 364B. Convex Optimization II. 3 Units.

Continuation of 364A. Subgradient, cutting-plane, and ellipsoid methods. Decentralized convex optimization via primal and dual decomposition. Monotone operators and proximal methods; alternating direction method of multipliers. Exploiting problem structure in implementation. Convex relaxations of hard problems. Global optimization via branch and bound. Robust and stochastic optimization. Applications in areas such as control, circuit design, signal processing, and communications. Course requirements include project. Prerequisite: 364A.

Same as: EE 364B

CME 369. Computational Methods in Fluid Mechanics. 3 Units.

The last two decades have seen the widespread use of Computational Fluid Dynamics (CFD) for analysis and design of thermal-fluids systems in a wide variety of engineering fields. Numerical methods used in CFD have reached a high degree of sophistication and accuracy. The objective of this course is to introduce ζ classical ζ approaches and algorithms used for the numerical simulations of incompressible flows. In addition, some of the more recent developments are described, in particular as they pertain to unstructured meshes and parallel computers. An in-depth analysis of the procedures required to certify numerical codes and results will conclude the course.

Same as: ME 469

CME 371. Computational Biology in Four Dimensions. 3 Units.

Cutting-edge research on computational techniques for investigating and designing the three-dimensional structure and dynamics of biomolecules, cells, and everything in between. These techniques, which draw on approaches ranging from physics-based simulation to machine learning, play an increasingly important role in drug discovery, medicine, bioengineering, and molecular biology. Course is devoted primarily to reading, presentation, discussion, and critique of papers describing important recent research developments. Prerequisite: CS 106A or equivalent, and an introductory course in biology or biochemistry. Recommended: some experience in mathematical modeling (does not need to be a formal course).

Same as: BIOMEDIN 371, BIOPHYS 371, CS 371

CME 372. Applied Fourier Analysis and Elements of Modern Signal Processing. 3 Units.

Introduction to the mathematics of the Fourier transform and how it arises in a number of imaging problems. Mathematical topics include the Fourier transform, the Plancherel theorem, Fourier series, the Shannon sampling theorem, the discrete Fourier transform, and the spectral representation of stationary stochastic processes. Computational topics include fast Fourier transforms (FFT) and nonuniform FFTs. Applications include Fourier imaging (the theory of diffraction, computed tomography, and magnetic resonance imaging) and the theory of compressive sensing. NOTE: Undergraduates require instructor permission to enroll. Undergraduates interested in taking the course should contact the instructor for permission, providing information about relevant background such as performance in prior coursework, reading, etc.

Same as: MATH 262

CME 375. Advanced Topics in Convex Optimization. 3 Units.

Modern developments in convex optimization: semidefinite programming; novel and efficient first-order algorithms for smooth and nonsmooth convex optimization. Emphasis on numerical methods suitable for large scale problems arising in science and engineering. Prerequisites: convex optimization (EE 364), linear algebra (Math 104), numerical linear algebra (CME 302); background in probability, statistics, real analysis and numerical optimization.

Same as: MATH 301

CME 390. Curricular Practical Training. 1 Unit.

Educational opportunities in high technology research and development labs in applied mathematics. Qualified ICME students engage in internship work and integrate that work into their academic program. Students register during the quarter they are employed and complete a research report outlining their work activity, problems investigated, results, and follow-on projects they expect to perform. May be repeated three times for credit.

CME 390A. Curricular Practical Training. 1 Unit.

Educational opportunities in high technology research and development labs in applied mathematics. Qualified ICME Ph.D. students engage in internship work and integrate that work into their academic program. Students register during the quarter they are employed and complete a research report outlining their work activity, problems investigated, results, and follow-on projects they expect to perform.

CME 399. Special Research Topics in Computational and Mathematical Engineering. 1-15 Unit.

Graduate-level research work not related to report, thesis, or dissertation. May be repeated for credit.

CME 400. Ph.D. Research. 1-15 Unit.

.

CME 444. Computational Consulting. 1-3 Unit.

Advice by graduate students under supervision of ICME faculty. Weekly briefings with faculty adviser and associated faculty to discuss ongoing consultancy projects and evaluate solutions. May be repeated for credit.

CME 500. Departmental Seminar. 1 Unit.

This seminar series in winter quarter will explore how ICME coursework and research is applied in various organizations around the world. It will feature speakers from ICME affiliate companies and ICME alumni giving technical talks on their use of computational math in their current roles. May be repeated for credit.

CME 801. TGR Project. 0 Units.

.

CME 802. TGR Dissertation. 0 Units.

.