COMPUTER SCIENCES (COMP SCI)

COMP SCI/L I S 102 — INTRODUCTION TO COMPUTING
3 credits.

Provides a broad overview of computing at an introductory level, including topics such as security, robotics, and artificial intelligence. Increases understanding of how computers work and how algorithms solve problems. Design and implement creative applications in an introductory coding environment. Provides a broad overview of computing and algorithms without an emphasis on programming.
Requisites: MATH 96 or placement into MATH 141. MATH 118 does not fulfill the prerequisite. Not open to students with credit for COMP SCI 300 or 320
Course Designation: Gen Ed - Quantitative Reasoning Part A
Breadth - Natural Science
Level - Elementary
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Develop a fundamental understanding of the key concepts of computer science in a variety of contexts.
Audience: Undergraduate

2. Create art, music, stories, games and other programs in a visual, introductory programming language
Audience: Undergraduate

3. Understand how computers use algorithms to solve problems and act in intelligent ways.
Audience: Undergraduate

4. Understand how computers utilize large sets of data to provide insight and knowledge.
Audience: Undergraduate

5. Learn how software and hardware make modern computers work
Audience: Undergraduate

COMP SCI 200 — PROGRAMMING I
3 credits.

Learn the process of incrementally developing small (200-500 lines) programs along with the fundamental Computer Science topics. These topics include: problem abstraction and decomposition, the edit-compile-run cycle, using variables of primitive and more complex data types, conditional and loop-based flow control, basic testing and debugging techniques, how to define and call functions (methods), and IO processing techniques. Also teaches and reinforces good programming practices including the use of a consistent style, and meaningful documentation. Intended for students who have no prior programming experience.
Requisites: Satisfied Quantitative Reasoning (QR) A requirement or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Gen Ed - Quantitative Reasoning Part B
Breadth - Natural Science
Level - Elementary
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Design and implement a standalone program that can interact with the user via prompts and or menus, access and edit data stored in an array or list structure, and use and further process the data found in those structures.
Audience: Undergraduate

2. Trace code to determine output or results.
Audience: Undergraduate

3. Implement a given program design and choose correct control structures for implementing algorithms expressed in pseudocode.
Audience: Undergraduate

4. Interpret a variety of diagram types used to express programming concepts and results: truth tables, memory model diagrams, control flow charts (activity diagrams), class diagrams, object diagrams, and use-case diagrams.
Audience: Undergraduate

5. List, describe, use the basic I/O operations for reading and writing text files to and from the computer's hard drive.
Audience: Undergraduate

6. Manipulate quantitative information to create models, and/or devise solutions to problems using multi-step arguments, based on and supported by quantitative information.
Audience: Undergraduate

7. Evaluate models and arguments using quantitative information.
Audience: Undergraduate

8. Express and interpret in context models, solutions and/or arguments using verbal, numerical, graphical algorithmic, computational or symbolic techniques.
Audience: Undergraduate

COMP SCI 220 – DATA SCIENCE PROGRAMMING I
4 credits.

Introduction to Data Science programming using Python. No previous programming experience required. Emphasis on analyzing real datasets in a variety of forms and visual communication.
Requisites: Satisfied Quantitative Reasoning (QR) A requirement or declared in the Professional Capstone Program in Computer Sciences. Not open to students with credit for COMP SCI 301.
Course Designation: Gen Ed - Quantitative Reasoning Part B
Breadth - Natural Science
Level - Elementary
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Integrate foundational concepts and tools from mathematics, computer science and statistics to solve data science problems
Audience: Undergraduate

2. Demonstrate competencies with tools and processes necessary for data management and reproducibility
Audience: Undergraduate

3. Produce meaning from data employing modeling strategies
Audience: Undergraduate

4. Demonstrate critical thinking related to data science concepts and methods
Audience: Undergraduate

5. Demonstrate oral, written and visual communication skills related to data science
Audience: Undergraduate

6. Manipulate quantitative information to create models, and/or devise solutions to problems using multi-step arguments, based on and supported by quantitative information
Audience: Undergraduate

COMP SCI/MATH 240 – INTRODUCTION TO DISCRETE MATHEMATICS
3 credits.

Basic concepts of logic, sets, partial order and other relations, and functions. Basic concepts of mathematics (definitions, proofs, sets, functions, and relations) with a focus on discrete structures: integers, bits, strings, trees, and graphs. Propositional logic, Boolean algebra, and predicate logic. Mathematical induction and recursion. Invariants and algorithmic correctness. Recurrences and asymptotic growth analysis. Fundamentals of counting.
Requisites: MATH 217 or 221
Course Designation: Breadth - Natural Science
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Demonstrate proficiency with basic skills for constructing and evaluating mathematically rigorous arguments and proofs.
Audience: Undergraduate

2. Construct proofs by induction to prove properties in a variety of domains (mathematical formulas, recursively-defined structures, loop invariants, program correctness).
Audience: Undergraduate

3. Demonstrate a familiarity with and an ability to reason about discrete structures/data types (integers, strings, bit strings, sets, relations, functions, graphs, trees).
Audience: Undergraduate

4. Employ basic program analysis techniques to evaluate behavior of functions and simple algorithms such as assessing correctness and termination of simple algorithms, constructing recurrences, solving simple recurrences, and applying asymptotic analysis.
Audience: Undergraduate

5. Apply basic combinatoric techniques to counting problems.
Audience: Undergraduate

6. Demonstrate the connections between a description of a simple language and a finite state machine or regular expression. Given a finite state machine or regular expression, determine the language it defines and/or whether a given string is recognized by it. Given a simple language, construct a finite state machine or regular expression that recognizes it.
Audience: Undergraduate

COMP SCI/E C E 252 — INTRODUCTION TO COMPUTER ENGINEERING
3 credits.

Logic components built with transistors, rudimentary Boolean algebra, basic combinational logic design, basic synchronous sequential logic design, basic computer organization and design, introductory machine- and assembly-language programming.
Requisites: None
Course Designation: Level - Elementary
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Perform basic operations on binary representations for data
Audience: Undergraduate

2. Analyze simple combinational and sequential digital logic and memory systems
Audience: Undergraduate

3. Identify the components and operation of an instruction set processor and write programs using assembly language
Audience: Undergraduate

4. Recognize and analyze ethical and professional responsibilities in engineering contexts
Audience: Undergraduate

COMP SCI 272 — INTRODUCTION TO WEB DEVELOPMENT
3 credits.

Introduces methods and tools for creating/maintaining secure and interactive web content. Topics include programming fundamentals to support core web concepts, application development essentials, and content management systems. Web best practices - such as accessibility, design, and critical thinking about relevant ethics and organization. Covers practical skills to design and implement websites using popular scripting languages and frameworks, content management systems (CMSs), and related tools.
Requisites: Not open to students with credit for L I S/COMP SCI 472.
Course Designation: Level - Elementary
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Develop understanding and application of current web scripting languages and development tools and frameworks.
Audience: Undergraduate

2. Install, configure, and customize open source content management systems.
Audience: Undergraduate

3. Understand and apply user experience and accessibility best practices in building accessible web-based systems.
Audience: Undergraduate

4. Design solutions to problems using multi-step scripting, logical operations, and functions.
Audience: Undergraduate

5. Understand ethical issues and concerns related to website development and its related technologies.
Audience: Undergraduate

6. Analyze the management challenges, and ethical considerations inherent in web development projects.
Audience: Undergraduate

COMP SCI 298 — DIRECTED STUDY IN COMPUTER SCIENCE
1-3 credits.

Undergraduate directed study in computer sciences.
Requisites: Consent of instructor
Course Designation: Level - Elementary
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 300 – PROGRAMMING II
3 credits.

Introduction to Object-Oriented Programming using classes and objects to solve more complex problems. Introduces array-based and linked data structures: including lists, stacks, and queues. Programming assignments require writing and developing multi-class (file) programs using interfaces, generics, and exception handling to solve challenging real world problems. Topics reviewed include reading/writing data and objects from/to files and exception handling, and command line arguments. Topics introduced: object-oriented design; class vs. object; create and define interfaces and iterators; searching and sorting; abstract data types (List,Stack,Queue,PriorityQueue(Heap),Binary Search Tree); generic interfaces (parametric polymorphism); how to design and write test methods and classes; array based vs. linked node implementations; introduction to complexity analysis; recursion.
Requisites: Satisfied QR-A and (COMP SCI 200, 220, 302, 310, 301, or placement into COMP SCI 300) or (E C E/COMP SCI 252 and E C E 203); graduate/professional standing; declared in Capstone Certificate in COMP SCI. Not open to students with credit for COMP SCI 367.
Course Designation: Gen Ed - Quantitative Reasoning Part B
Breadth - Natural Science
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. List and describe common operations for List, Stack, Queue, Priority Queue, Tree.
Audience: Undergraduate

2. Analyze the time-complexity and compare the Big-Oh O(n) worse case complexity of different ADT implementations, and the complexity for any data structures and algorithms used to implement those operations.
Audience: Undergraduate

3. Identify and properly test all boundary conditions for comprehensive testing of their programs.
Audience: Undergraduate

4. Interpret and create a variety of diagrams: Call Stack trace; Stack, Queue, and Tree Data Structures; Recursive call (list or tree), control flow charts, class diagrams, object diagrams, and use-case diagrams.
Audience: Undergraduate

5. Implement Object-Oriented (multi-class) standalone programs that manage a variety of data storage and retrieval operations (Program development skills and experience).
Audience: Undergraduate

6. Evaluate models and arguments using quantitative information.
Audience: Undergraduate

COMP SCI 304 – PEER COLLABORATION IN COMPUTER SCIENCES (WES-CS)
1 credit.

Interactive opportunity to discuss basic computer science concepts such as programming, including data structures, algorithms, object-oriented programming principles and techniques for efficient coding, testing, and debugging in a smaller setting with peers. Requires concurrent enrollment in COMP SCI 200, 220, 252, 300, 320, 354, or 400.
Requisites: Consent of instructor
Course Designation: Level - Elementary
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026
Learning Outcomes: 1. Demonstrate collaborative programming and thinking skills in group settings.
Audience: Undergraduate

2. Explain computer science concepts verbally.
Audience: Undergraduate

3. Describe computing's impact on communities, including its role in perpetuating bias and power.
Audience: Undergraduate

COMP SCI 310 – PROBLEM SOLVING USING COMPUTERS
3 credits.

Gives students an introduction to computer and analytical skills to use in their subsequent course work and professional development. Discusses several methods of using computers to solve problems, including elementary programming techniques, symbolic manipulation languages, and software packages. Techniques will be illustrated using sample problems drawn from elementary engineering. Emphasis is on introduction of algorithms with the use of specific tools to illustrate the methods.
Requisites: MATH 222, graduate/professional standing, or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Elementary
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2022

COMP SCI 319 – DATA SCIENCE PROGRAMMING I FOR RESEARCH
3 credits.

Introduction to Data Science programming using Python. In addition to a survey of programming basics (control flow and data structures), web scraping, database queries, and tabular analysis will be introduced. Projects will emphasize analyzing real datasets in a variety of forms and visual communication using plotting tools. Similar to COMP SCI 220 but the pedagogical style of the projects will be adapted to graduate students in fields other than computer science and data science. No previous programming experience required.
Requisites: Graduate/professional standing
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. integrate foundational concepts and tools from mathematics, computer science and statistics to solve data science problems
Audience: Undergraduate

2. demonstrate competencies with tools and processes necessary for data management and reproducibility
Audience: Undergraduate

3. produce meaning from data employing modeling strategies
Audience: Undergraduate

4. demonstrate critical thinking related to data science concepts and methods
Audience: Undergraduate

5. demonstrate oral, written and visual communication skills related to data science
Audience: Undergraduate

6. manipulate quantitative information to create models, and/or devise solutions to problems using multi-step arguments, based on and supported by quantitative information
Audience: Undergraduate

COMP SCI 320 – DATA SCIENCE PROGRAMMING II
4 credits.

Intermediate approach to Data Science programming using Python. Experience with basic tabular analysis in Python is assumed. Learn to implement data structures (e.g., graphs) to efficiently represent datasets. Software-engineering tools such as version control and Python virtual environments will be introduced, with an emphasis on reproducibility of analysis. Tracing and A/B testing will be introduced as techniques for generating meaningful datasets. Introduces basic classification, clustering, optimization, and simulation techniques. Plotting and visual communication will be emphasized throughout the course.
Requisites: COMP SCI 220 (or COMP SCI 301 prior to Spring 2020), COMP SCI 300, 319, graduate/professional standing, or declared in the Computer Sciences for Professionals Capstone Certificate
Course Designation: Breadth - Natural Science
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. integrate foundational concepts and tools from mathematics, computer science and statistics to solve data science problems
Audience: Undergraduate

2. demonstrate competencies with tools and processes necessary for data management and reproducibility
Audience: Undergraduate

3. produce meaning from data employing modeling strategies
Audience: Undergraduate

4. demonstrate critical thinking related to data science concepts and methods
Audience: Undergraduate

5. conduct data science activities aware of and according to policy, privacy, security and ethical considerations
Audience: Undergraduate

6. demonstrate oral, written and visual communication skills related to data science
Audience: Undergraduate

COMP SCI/E C E 352 – DIGITAL SYSTEM FUNDAMENTALS
3 credits.

Logic components, Boolean algebra, combinational logic analysis and synthesis, synchronous and asynchronous sequential logic analysis and design, digital subsystems, computer organization and design.
Requisites: Satisfied Quantitative Reasoning (QR) A requirement and E C E/COMP SCI 252
Course Designation: Gen Ed - Quantitative Reasoning Part B
Breadth - Physical Sci. Counts toward the Natural Sci req
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Perform operations on signed and unsigned numbers, including evaluating overflow
Audience: Undergraduate

2. Implement Boolean logic circuits, use Boolean identities to perform algebraic manipulations, use Karnaugh maps to implement any function of 4 variables, use DeMorgans's Theorem, implement any function as SOP or POS
Audience: Undergraduate

3. Design datapath circuits, decoders, muxes, priority encoders, tri-states, understand hierarchy and how to build up larger datapaths from blocks, design ALUs and other digital circuits using rudimentary HDL constructs
Audience: Undergraduate

4. Design sequential circuits, analyze synchronous vs. asynchronous designs and flip-flops vs. latches, trace the behavior of a sequential circuit, design state machines for control Logic
Audience: Undergraduate

5. Analyze basic processor architecture, define a control word and analyze operation of the datapath in relation to it, describe the basic operation of a single-cycle stored program computer
Audience: Undergraduate

COMP SCI/E C E 354 – MACHINE ORGANIZATION AND PROGRAMMING
3 credits.

An introduction to fundamental structures of computer systems and the C programming language with a focus on the low-level interrelationships and impacts on performance. Topics include the virtual address space and virtual memory, the heap and dynamic memory management, the memory hierarchy and caching, assembly language and the stack, communication and interrupts/signals, compiling and assemblers/linkers.
Requisites: E C E/COMP SCI 252 and (COMP SCI 300 or 302) or graduate/professional standing or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Gen Ed - Quantitative Reasoning Part B
Breadth - Natural Science
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Implement and interpret C programs using standard tools, and relate C language constructs to both assembly language and fundamental computer system structures.
Audience: Undergraduate

2. Differentiate, describe, and diagram the memory segments of a process's virtual address space and explain how each is used in C programs.
Audience: Undergraduate

3. Describe and diagram how dynamically allocated (heap) memory works, analyze the performance of allocation strategies, and implement a dynamic memory manager in C.
Audience: Undergraduate

4. Differentiate memory hierarchy levels, relative performance, and space differences, differentiate between common cache configurations, and appraise the effects of data structures and access patterns on spatial and temporal locality and available memory caching on the performance of C programs.
Audience: Undergraduate

5. Diagram a stack trace of execution for function calls in C programs, and explain how the compiler implements the stack with assembly language code.
Audience: Undergraduate

6. Write C programs that send and receive signals and respond to exceptional circumstances, and explain the underlying mechanism that enables asynchronous execution.
Audience: Undergraduate

7. Identify and summarize the steps required to build an executable program from multiple C source code files and compiled libraries, and describe the processes of linking multiple object code modules to form an executable and loading an executable to run it.
Audience: Undergraduate

COMP SCI 368 – LEARNING A PROGRAMMING LANGUAGE
1 credit.

For students interested in learning a particular programming language. Focuses on a specific language offered at one of three levels: beginner, intermediate, and advanced. Students may repeat the course if the topic title is different.
Requisites: None
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 400 – PROGRAMMING III
3 credits.

The third course in our programming fundamentals sequence. Covers balanced search trees, graphs, graph traversal algorithms, hash tables and sets, and complexity analysis and about classes of problems that require each data type. Involves design and implementation using high quality professional code, a medium sized program, that demonstrates knowledge and use of latest language features, tools, and conventions. Additional topics may include inheritance and polymorphism; anonymous inner classes, lambda functions, performance analysis to discover and optimize critical code blocks. Introduces industry standards for writing high-quality, maintainable code.
Requisites: COMP SCI 300, graduate/professional standing, or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Breadth - Natural Science
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Select and use standard development tools, including shells, source control and build systems.
Audience: Undergraduate

2. Develop a program including making and describing high and low-level design decisions.
Audience: Undergraduate

3. Apply, implement, and analyze sorting, tree and graph algorithms.
Audience: Undergraduate

4. Select and apply common indexing data structures to represent sets and maps.
Audience: Undergraduate

5. Design and categorize different types of tests and pick types based on the testing scenario.
Audience: Undergraduate

6. Identify and solve problems by applying standard methods including graph algorithms, balanced trees, and sorting.
Audience: Undergraduate

7. Read, analyze and review programs to summarize and evaluate their functionality, performance and maintainability.
Audience: Undergraduate

COMP SCI/L I S/STAT 401 – UNDERGRADUATE COOPERATIVE EDUCATION
1 credit.

Full time work experience which combines classroom theory with practical knowledge related to Computer Sciences, Data Science, Statistics, or Information Science.
Requisites: Consent of instructor
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Workplace - Workplace Experience Course
Repeatable for Credit: Yes, for 3 number of completions
Last Taught: Spring 2026
Learning Outcomes: 1. Apply academic experience gained through coursework in a professional setting.
Audience: Undergraduate

2. Experience the nature and demands of a professional career in computer science, information science, and/or statistics/data science
Audience: Undergraduate

3. Develop professional and transferable skills like time management, collaboration, problem-solving, and communication in the workplace.
Audience: Undergraduate

COMP SCI 402 – INTRODUCING COMPUTER SCIENCE TO K-12 STUDENTS
2 credits.

Work in teams to lead Computer Science clubs and workshops for K-12 students at sites in the Madison area. Design and lead activities to help K-12 students learn computational thinking and computer programming.
Requisites: COMP SCI 200, 220, 300, 301, 302, 310, 367, placement into COMP SCI 300, or L I S/COMP SCI 102 (COMP SCI 202 prior to Fall 2023), graduate/professional standing, or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Write programs in the Scratch programming language.
Audience: Undergraduate

2. Design and lead after-school programming clubs for K-12 students.
Audience: Undergraduate

3. Incorporate effective instructional strategies when teaching K-12 students.
Audience: Undergraduate

4. Reflect on and document their teaching.
Audience: Undergraduate

COMP SCI/STAT 403 – INTERNSHIP COURSE IN COMP SCI AND DATA SCIENCE
1 credit.

Enables students with outside internships to earn academic credit connected to their work experience related to the Computer Sciences or Data Science programs.
Requisites: Consent of instructor
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, for 3 number of completions
Last Taught: Spring 2026
Learning Outcomes: 1. Understand the challenges and opportunities in Computer Sciences and Data Science professions
Audience: Undergraduate

2. Be prepared to find, apply and interview for a job and/or additional education
Audience: Undergraduate

3. Articulate your career goals and long-term trajectory
Audience: Undergraduate

COMP SCI 407 — FOUNDATIONS OF MOBILE SYSTEMS AND APPLICATIONS
3 credits.

Design and implementation of applications, systems, and services for mobile platforms with (i) constraints, such as limited processing, memory, energy, interfaces, variable bandwidth, and high mobility, and (ii) features, such as touchscreens, cameras, electronic compasses, GPS, and accelerometers.
Requisites: COMP SCI 400 or graduate/professional standing
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Understand the mobile ecosystem.
Audience: Undergraduate

2. Understand basics of mobile system design and constituents, including wireless communication, location services, energy management, use of cloud services, and mobile device interface design.
Audience: Undergraduate

3. Demonstrate the fundamentals of mobile app development: setting up the development environment, creating an app from scratch, and understanding the app lifecycle.
Audience: Undergraduate

4. Understand the various app components, strategies for optimizing app performance, and use of mobile features.
Audience: Undergraduate

5. Construct end-to-end applications that use server backends.
Audience: Undergraduate

6. Demonstrate strong teamwork capabilities including collaborative working, task distribution, and conflict resolution.
Audience: Undergraduate

COMP SCI 412 — INTRODUCTION TO NUMERICAL METHODS
3 credits.

Interpolation, solution of linear and nonlinear systems of equations, approximate integration and differentiation, numerical solution of ordinary differential equations, Data fitting (such as least squares) by polynomials and splines. Knowledge of matrix algebra recommended, such as MATH 340.
Requisites: MATH 222 and (MATH/COMP SCI 240 or MATH 234) and (COMP SCI 200, 300, 301, 302, 310, or placement into COMP SCI 300) or graduate/professional standing or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Breadth - Natural Science
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026

COMP SCI/I SY E/MATH 425 — INTRODUCTION TO COMBINATORIAL OPTIMIZATION
3 credits.

Focuses on optimization problems over discrete structures, such as shortest paths, spanning trees, flows, matchings, and the traveling salesman problem. We will investigate structural properties of these problems, and we will study both exact methods for their solution, and approximation algorithms.
Requisites: (MATH 320, 340, 341, or 375) or graduate/professional standing or member of the Pre-Masters Mathematics (Visiting International) Program
Course Designation: Breadth - Physical Sci. Counts toward the Natural Sci req
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Identify and use the structural properties of combinatorial optimization problems
Audience: Undergraduate

2. Apply algorithms for the solution -exact or approximate- of a combinatorial optimization problem
Audience: Undergraduate

3. Explain why the algorithms studied are correct and understand their running time
Audience: Undergraduate

COMP SCI/E C E/MATH 435 – INTRODUCTION TO CRYPTOGRAPHY

3 credits.

Cryptography is the art and science of transmitting digital information in a secure manner. Provides an introduction to its technical aspects.
Requisites: (MATH 320, 340, 341, or 375) or graduate/professional standing or member of the Pre-Masters Mathematics (Visiting International) Program
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Describe and distinguish between perfect secrecy and computational security.
Audience: Undergraduate

2. State game#based security definitions—IND#CPA, IND#CCA, MAC unforgeability, and signature unforgeability—in both symmetric# and public#key contexts.
Audience: Undergraduate

3. Explain core primitives: pseudorandom functions, block ciphers, and collision#resistant hash functions.
Audience: Undergraduate

4. Contrast foundational hardness assumptions: one#way functions, PRFs, Diffie–Hellman, discrete logarithm, and integer factoring.
Audience: Undergraduate

5. Construct symmetric and public#key encryption and authentication schemes using these primitives, and prove their security.
Audience: Undergraduate

6. Evaluate the security and deployment trade#offs of protocols such as authenticated encryption, key exchange, and digital signatures.
Audience: Undergraduate

COMP SCI/STAT 471 – INTRODUCTION TO COMPUTATIONAL STATISTICS

3 credits.

Classical statistical procedures arise where closed-form mathematical expressions are available for various inference summaries (e.g. linear regression; analysis of variance). A major emphasis of modern statistics is the development of inference principles in cases where both more complex data structures are involved and where more elaborate computations are required. Topics from numerical linear algebra, optimization, Monte Carlo (including Markov chain Monte Carlo), and graph theory are developed, especially as they relate to statistical inference (e.g., bootstrapping, permutation, Bayesian inference, EM algorithm, multivariate analysis).
Requisites: STAT/MATH 310 and (STAT 333 or 340), graduate/professional standing, or declared in Statistics VISP
Course Designation: Breadth - Natural Science
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Use computational tools (alongside mathematical ones) to extract information from (a) the likelihood function, the central object of interest in frequentist statistics, and (b) the posterior distribution, the central object of interest in Bayesian statistics
Audience: Undergraduate

2. Describe, understand the theoretical properties of, and implement basic algorithms for optimizing likelihood functions, including least squares and the IRLS algorithm, and the EM algorithm
Audience: Undergraduate

3. Understand random numbers and pseudorandom numbers and how to distinguish them, and utilize a variety of techniques for generating random variates from a probability distribution
Audience: Undergraduate

4. Use Monte Carlo methodology for such purposes as (a) carrying out a simulation study to study the properties of a statistical method, or (b) performing statistical inference via the bootstrap, or MCMC
Audience: Undergraduate

5. Understand the use of graphical models for representing the structure of complex joint distributions, and be able to use computational tools to extract information from graphical models
Audience: Undergraduate

COMP SCI/L I S 472 – INTRODUCTION TO WEB DEVELOPMENT
3 credits.

Applied web development introduces methods and tools for creating/maintaining secure and interactive web content. Topics include programming fundamentals to support core web concepts, application development essentials, and content management systems. Web best practices - such as accessibility, design, and critical thinking about relevant ethics and organization - will be incorporated throughout. Gain practical skills to design and implement websites using popular scripting languages and frameworks, content management systems (CMSs), and related tools.
Requisites: Junior standing, declared in Library and Information Studies MA, Information MS, or Capstone Certificate in Computer Sciences for Professionals. Not open to students with credit for COMP SCI 272.
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Develop understanding and application of current web scripting languages and development tools and frameworks.
Audience: Both Grad & Undergrad

2. Install, configure, and customize open source content management systems.
Audience: Both Grad & Undergrad

3. Understand and apply user experience and accessibility best practices in building accessible websites.
Audience: Both Grad & Undergrad

4. Design solutions to problems using multi-step scripting, logical operations, and functions.
Audience: Both Grad & Undergrad

5. Understand ethical issues and concerns related to website development and its related technologies.
Audience: Both Grad & Undergrad

6. Analyze the management challenges, and ethical considerations inherent in web development projects.
Audience: Both Grad & Undergrad

7. Critically evaluate and compare different frameworks and libraries for extending scripting capabilities.
Audience: Graduate

COMP SCI/MATH/STAT 475 – INTRODUCTION TO COMBINATORICS
3 credits.

Problems of enumeration, distribution, and arrangement. Inclusion-exclusion principle. Generating functions and linear recurrence relations. Combinatorial identities. Graph coloring problems. Finite designs. Systems of distinct representatives and matching problems in graphs. Potential applications in the social, biological, and physical sciences. Puzzles. Problem solving.
Requisites: (MATH 320, 340, 341, or 375) or graduate/professional standing or member of the Pre-Masters Mathematics (Visiting International) Program
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Understand basic counting strategies, such as staged thought-experiments, inclusion/exclusion, generating functions, and recurrence relations, and apply these strategies to solve a wide variety of counting problems.
Audience: Undergraduate

2. Recall basic objects that are used in combinatorics, such as permutations and combinations of sets and multisets, binomial and multinomial coefficients, the Catalan numbers, the Stirling numbers, and the partition numbers.
Audience: Undergraduate

3. Analyze a given combinatorial problem using the standard theorems of combinatorics, such as the pigeonhole principle, the Newton binomial theorem, the multinomial theorem, the Ramsey theorem, the Dilworth theorem, the Burnside theorem, and the Polya counting theorem.
Audience: Undergraduate

4. Construct mathematical arguments related to combinatorial problems using the above definitions, properties, theorems, and counting strategies; including the construction of examples and counterexamples.
Audience: Undergraduate

5. Convey his or her arguments in oral and written form in English, using appropriate mathematical terminology, notation, and grammar.
Audience: Undergraduate

COMP SCI/CURRIC 502 – THEORY AND PRACTICE IN COMPUTER SCIENCE EDUCATION
1 credit.

Computer science educational pedagogy and general teaching practices. Practical experience gained through tutoring students. Knowledge of object-oriented programming required.
Requisites: COMP SCI 300 or 302 or declared in Computer Science graduate program
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Describe and demonstrate best practices in the teaching of computer science.
Audience: Undergraduate

2. Create learning materials that exemplify selected computer science pedagogical topics.
Audience: Undergraduate

3. Identify effective tutoring session strategies.
Audience: Undergraduate

COMP SCI/E C E 506 – SOFTWARE ENGINEERING
3 credits.

Ideas and techniques for designing, developing, and modifying large software systems. Topics include software engineering processes; requirements and specifications; project team organization and management; software architectures; design patterns; testing and debugging; and cost and quality metrics and estimation. Students will work in large teams on a substantial programming project.
Requisites: (COMP SCI 367 or 400) and (COMP SCI 407, 536, 537, 545, 559, 564, 570, 679 or E C E/COMP SCI 552) or graduate/professional standing, or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Execute a significant software development project within a team
Audience: Undergraduate

2. Use software tools for collaboration including source control, project management, and automated testing
Audience: Undergraduate

3. Collaborate effectively on a significant software development project as part of a team, demonstrating professionalism in communication, accountability, and teamwork.
Audience: Undergraduate

4. Describe and participate in various SWE inspection activities (e.g. Group Code Review; Pair Programming; Asynchronous Review).
Audience: Undergraduate

5. Characterize and contrast software engineering methodologies and development techniques, including Agile, Waterfall, DevOps, Test-driven Development, Comment-First Development, and Extreme Programming
Audience: Undergraduate

6. Characterize and contrast software design philosophies and principles.
Audience: Undergraduate

7. Explain and evaluate techniques for software deployment such as Container and VM technologies.
Audience: Undergraduate

COMP SCI/MATH 513 – NUMERICAL LINEAR ALGEBRA
3 credits.

Direct and iterative solution of linear and nonlinear systems and of eigenproblems. LU and symmetric LU factorization. Complexity, stability, and conditioning. Nonlinear systems. Iterative methods for linear systems. QR-factorization and least squares. Eigenproblems: local and global methods.
Requisites: (MATH 340, 341, or 375) and (COMP SCI 200, 300, 301, 302, 310, or placement into COMP SCI 300) or graduate/professional standing or member of the Pre-Masters Mathematics (Visiting International) program
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026

COMP SCI/MATH 514 – NUMERICAL ANALYSIS
3 credits.

Polynomial forms, divided differences. Polynomial interpolation. Polynomial approximation: uniform approximation and Chebyshev polynomials, least-squares approximation and orthogonal polynomials. Numerical differentiation and integration. Splines, B-splines and spline approximation. Numerical methods for solving initial and boundary value problems for ordinary differential equations.
Requisites: (MATH 320, 340, 341, or 375), (MATH 322, 376, 421, or 521), and (COMP SCI 200, 220, 300, 310, or 301 prior to Spring 2020, or placement into COMP SCI 300); grad/professional standing; member of the Pre-Masters Mathematics (Visiting International) Program
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Recall and state the formal definitions of the mathematical objects and their properties used in numerical analysis (e.g., Lagrange polynomials, Gibbs phenomenon, Runge phenomenon, orthogonal polynomials, recurrence relation, Gaussian quadrature points, splines, etc.).
Audience: Both Grad & Undergrad

2. Use different techniques of numerical analysis in their appropriate settings (e.g., polynomial interpolation, least square approximation, discrete Fourier transform, the Golub-Welsch algorithm, fast Fourier transform, trapezoidal rule and Simpson's rule, numerical differentiation, forward and backward Euler's method, etc.).
Audience: Both Grad & Undergrad

3. State the main theoretical results related to the error analysis for different methods (e.g., least square error, numerical integration using a Riemann sum, the trapezoidal rule, Simpson's rule and Gaussian quadratures, (semi-)discrete Fourier transform, forward and backward Euler, etc.), and recall the arguments for these theorems and the underlying logic of their proofs.
Audience: Both Grad & Undergrad

4. Convey arguments in oral and written forms using English and appropriate mathematical terminology, notation and grammar.
Audience: Both Grad & Undergrad

5. Identify applications of course content in current areas of research.
Audience: Graduate

COMP SCI/DS/I SY E 518 – WEARABLE TECHNOLOGY
3 credits.

Gives students hands-on experience in building wearable computing platforms. Designed for students who have a background in textiles and apparel design, computer science, engineering or media arts. By the completion of the course students will have fundamental knowledge of electronic circuitry, programming, and "maker skills".
Requisites: Sophomore standing
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2022

COMP SCI 520 – INTRODUCTION TO THEORY OF COMPUTING
3 credits.

Basics about the notion, capabilities, and limitations of computation: elements of finite automata and regular languages, computability theory, and computational complexity theory. Additional topics include context-free grammars and languages, and complexity-theoretic cryptography.
Requisites: (MATH/COMP SCI 240 or STAT/COMP SCI/MATH 475) and (COMP SCI 367 or 400), or graduate/professional standing, or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Design and implement deterministic finite automata to capture specified regular languages.
Audience: Undergraduate

2. Differentiate between deterministic and nondeterministic finite automata and apply finite automata theory to express those computation tasks that can be captured by regular languages.
Audience: Undergraduate

3. Express and interpret computational models that lead to and capture algorithmic thinking.
Audience: Undergraduate

4. Design and implement a pushdown automata to capture specified context-free languages.
Audience: Undergraduate

5. Identify and explain reasoning about formalization of computation, formulated as various models of computation and make transformations between finite automata and regular languages, pushdown automata and context-free languages.
Audience: Undergraduate

6. Construct clear, concise, and sound proofs for statements regarding different models of computation, their relative strengths and weaknesses.
Audience: Undergraduate

7. Reason with Turing machines, and the various equivalent formulations of computability theory, for example to explain how the formalization captures everyday computational processes.
Audience: Undergraduate

8. Reason about computational intractability, applying the concepts of NP-hardness and NP-completeness to classify computational problems.
Audience: Undergraduate

COMP SCI/E C E/I SY E 524 – INTRODUCTION TO OPTIMIZATION
3 credits.

Introduction to mathematical optimization from a modeling and solution perspective. Formulation of applications as discrete and continuous optimization problems and equilibrium models. Survey and appropriate usage of basic algorithms, data and software tools, including modeling languages and subroutine libraries.
Requisites: (COMP SCI 200, 220, 300, 301, 302, 310, or placement into COMP SCI 300) and (MATH 320, 340, 341, or 375) or graduate/professional standing
Course Designation: Breadth - Natural Science
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Engage in topics about "optimization in practice".
Audience: Undergraduate

2. Use and analyze the results of state of the art optimization software.
Audience: Undergraduate

3. Use the GAMS modeling system and Jupyter notebooks (in conjunction with elementary Python) or Julia and JUMP.
Audience: Undergraduate

4. Design good models for realistic applications in engineering and the sciences.
Audience: Undergraduate

5. Develop a "commercial strength" application of optimization technology.
Audience: Undergraduate

COMP SCI/I SY E/MATH/STAT 525 – LINEAR OPTIMIZATION
3 credits.

Introduces optimization problems whose constraints are expressed by linear inequalities. Develops geometric and algebraic insights into the structure of the problem, with an emphasis on formal proofs. Presents the theory behind the simplex method, the main algorithm used to solve linear optimization problems. Explores duality theory and theorems of the alternatives.
Requisites: MATH 320, 340, 341, 375, or 443 or graduate/professional standing or member of the Pre-Masters Mathematics (Visiting International) Program
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Use linear programming to formulate real world decision problems.
Audience: Both Grad & Undergrad

2. Apply algorithms to solve linear programming problems and demonstrate their correctness.
Audience: Both Grad & Undergrad

3. Combine different proving techniques explored in class in an original way to show new results.
Audience: Graduate

COMP SCI/I SY E 526 — ADVANCED LINEAR PROGRAMMING
3 credits.

Review of linear programming. Polynomial time methods for linear programming. Quadratic programs and linear complementarity problems and related solution techniques. Solution sets and their continuity properties. Error bounds for linear inequalities and programs. Parallel algorithms for linear and quadratic programs.
Requisites: STAT/COMP SCI/I SY E/MATH 525 and (COMP SCI 200, 220, 300, 301, 302, 310, or placement into COMP SCI 300) or graduate/professional standing
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2020
Learning Outcomes: 1. Use the theory of linear programming to prove general duality results
Audience: Undergraduate

2. Apply the concept of complementarity
Audience: Undergraduate

3. Analyze and develop algorithms for solving optimization and equilibrium problems
Audience: Undergraduate

4. Apply decomposition methods and other advanced algorithms for the solution of optimization and equilibrium problems
Audience: Undergraduate

5. Understand economic concepts and how they relate to optimization and equilibria
Audience: Undergraduate

6. Extend theory of linear programming into the framework of conic programming
Audience: Undergraduate

COMP SCI/E C E/M E 532 — MATRIX METHODS IN MACHINE LEARNING
3 credits.

Linear algebraic foundations of machine learning featuring real-world applications of matrix methods from classification and clustering to denoising and data analysis. Mathematical topics include: linear equations, regression, regularization, the singular value decomposition, and iterative algorithms. Machine learning topics include: the lasso, support vector machines, kernel methods, clustering, dictionary learning, neural networks, and deep learning. Previous exposure to numerical computing (e.g. Matlab, Python, Julia, R) required.
Requisites: (MATH 234, 320, 340, 341, or 375) and (E C E 203, COMP SCI 200, 220, 300, 301, 302, 310, 320, or placement into COMP SCI 300), graduate/professional standing, or declared in Capstone Certificate in Computer Sciences for Professionals
Course Designation: Breadth - Physical Sci. Counts toward the Natural Sci req
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Use matrices and vectors to formulate classification, prediction and matrix completion problems using techniques such as least squares, regularized least squares, the singular value decomposition, subspace methods, support vector machines, neural networks and kernel methods.
Audience: Both Grad & Undergrad

2. Implement machine learning techniques for classification, prediction and matrix completion problems in software, and validate their performance on datasets using cross validation.
Audience: Both Grad & Undergrad

3. Apply advanced techniques to formulate and prove optimality of various matrix based techniques in machine learning.
Audience: Graduate

COMP SCI/E C E 533 – IMAGE PROCESSING
3 credits.

Mathematical representation of continuous and digital images; models of image degradation; picture enhancement, restoration, segmentation, and coding; pattern recognition, tomography.
Requisites: E C E 330 and (MATH 320 or 340), graduate/professional standing, or member of Engineering Guest Students
Course Designation: Breadth - Physical Sci. Counts toward the Natural Sci req
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Process digital images using available engineering software
Audience: Both Grad & Undergrad

2. Use time-domain and frequency-domain methods to analyze images and their properties
Audience: Both Grad & Undergrad

3. Apply nonlinear filters to images such as morphological operations, edge-preserving nonlinearities, and optimization-based filters
Audience: Both Grad & Undergrad

4. Segment images using both small-scale and large-scale techniques
Audience: Both Grad & Undergrad

5. Apply classification techniques to image recognition
Audience: Both Grad & Undergrad

6. Extract features from images and apply to tasks such as registration and super-resolution
Audience: Both Grad & Undergrad

7. Lead team in meeting objectives of image processing operations
Audience: Graduate

COMP SCI 536 – INTRODUCTION TO PROGRAMMING LANGUAGES AND COMPILERS
3 credits.

Introduction to the theory and practice of compiler design. Comparison of features of several programming languages and their implications for implementation techniques. Several programming projects required.
Requisites: E C E/COMP SCI 354 and (COMP SCI 367 or 400) or graduate/professional standing or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Construct and manipulate abstract syntax trees (ASTs) to represent the structure of source programs, applying grammar rules and language constructs.
Audience: Undergraduate

2. Compare and contrast interpreters and compilers, focusing on AST passes and execution models.
Audience: Undergraduate

3. Design and implement interpreters by traversing and evaluating ASTs for simple and structured programming languages
Audience: Undergraduate

4. Develop lexers and parsers that convert textual program input into ASTs, using formal grammar specifications and parsing techniques.
Audience: Undergraduate

5. Implement static analyses, such as type checking, by analyzing and annotating ASTs to enforce program correctness rules.
Audience: Undergraduate

6. Translate ASTs into low-level code (e.g., x86 assembly), applying principles of code generation and basic optimization.
Audience: Undergraduate

7. Analyze how advanced language features, such as classes, objects, and garbage collection, introduce complexity in interpretation and compilation.

Audience: Undergraduate

COMP SCI 537 – INTRODUCTION TO OPERATING SYSTEMS
4 credits.

Input-output hardware, interrupt handling, properties of magnetic tapes, discs and drums, associative memories and virtual address translation techniques. Batch processing, time sharing and real-time systems, scheduling resource allocation, modular software systems, performance measurement and system evaluation.
Requisites: E C E/COMP SCI 354 and (COMP SCI 367 or 400) or graduate/professional standing or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Describe the role and core responsibilities of an operating system
Audience: Undergraduate

2. Explain and evaluate key OS abstractions and mechanisms
Audience: Undergraduate

3. Analyze and compare scheduling, memory management, and storage strategies
Audience: Undergraduate

4. Implement and extend functionality in an operating system
Audience: Undergraduate

5. Apply systems-level thinking to reason about performance, reliability, and correctness
Audience: Undergraduate

6. Write and debug multi-threaded programs
Audience: Undergraduate

COMP SCI 538 – INTRODUCTION TO THE THEORY AND DESIGN OF PROGRAMMING LANGUAGES
3 credits.

Design and theory of programming languages: procedural, object-oriented, functional and logic paradigms. Serial and concurrent programming. Execution models and formal specification techniques.
Requisites: E C E/COMP SCI 354 and (COMP SCI 367 or 400) or graduate/professional standing or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Read, write, and transform functional programs using key paradigms such as lazy evaluation, eager evaluation, and monads.
Audience: Undergraduate

2. Apply lambda calculus as a formal foundation for understanding and reasoning about functional programming languages.
Audience: Undergraduate

3. Reason about and analyze program behavior through type checking, type inference, and static analysis techniques.
Audience: Undergraduate

4. Construct and reason about control flow in programs using techniques such as continuation-passing style (CPS) and monadic control structures.
Audience: Undergraduate

5. Evaluate the role of type systems, polymorphism, and abstraction mechanisms in enforcing program correctness and expressiveness.
Audience: Undergraduate

6. Analyze how programming language features influence program security, with a focus on reasoning about memory management and control flow vulnerabilities.
Audience: Undergraduate

7. Apply formal reasoning techniques to construct correctness proofs of programs.
Audience: Undergraduate

COMP SCI/E C E/M E 539 — INTRODUCTION TO ARTIFICIAL NEURAL NETWORKS
3 credits.

Theory and applications of artificial neural networks: multi-layer perceptron, self-organization mapdeep neural network convolutional neural network, recurrent network, support vector machines genetic algorithm, and evolution computing. Applications to control, pattern recognition, prediction, and object detection and tracking.
Requisites: COMP SCI 200, 220, 300, 301, 302, 310, placement into COMP SCI 300, or graduate/professional standing
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Identify if a given data analysis task is a pattern classification problem or a model approximation problem.
Audience: Undergraduate

2. Apply multi-layer perceptron neural network training algorithm to develop artificial neural network (ANN) based pattern classifiers and data predictors.
Audience: Undergraduate

3. Apply deep learning network for pattern classification
Audience: Undergraduate

4. Apply support vector machine (SVM) to develop pattern classifiers.
Audience: Undergraduate

5. Apply self-organization map and k-means to perform clustering operations of a given data set.
Audience: Undergraduate

6. Apply stochastic optimization methods, including simulated annealing, genetic algorithm and random search to solve a discrete optimization problem.
Audience: Undergraduate

COMP SCI 540 — INTRODUCTION TO ARTIFICIAL INTELLIGENCE
3 credits.

Principles of knowledge-based search techniques, automatic deduction, knowledge representation using predicate logic, machine learning, probabilistic reasoning. Applications in tasks such as problem solving, data mining, game playing, natural language understanding, computer vision, speech recognition, and robotics.
Requisites: (COMP SCI 300, 320 or 367) and (MATH 211, 217, 221, or 275) or graduate/professional standing or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. (Uninformed Search Methods) Identify the formulation of search for problem solving tasks. Understand important concepts in uninformed search. Apply the search methods on the formulated search problem.
Audience: Undergraduate

2. (Informed Search Methods) Understand important concepts in informed search. Differentiate from uninformed search. Solve the formulated search problem with the informed search method A*.
Audience: Undergraduate

3. (Local Search Methods) Identify the formulation of search for problem solving tasks. Apply the hill climbing method for local search problems. Identify and summarize the important features of the simulated annealing and genetic algorithms.
Audience: Undergraduate

4. (Game Playing) Recall the concept of games. Perform the minimax game playing method on formulated game tasks. Apply alpha-beta pruning to speed up the minimax method.
Audience: Undergraduate

5. (Unsupervised and Supervised Learning) Identify and summarize important features about supervised learning and unsupervised learning. Differentiate between the two types of tasks.
Audience: Undergraduate

6. (Classic Learning Methods) Apply linear regression, hierarchical agglomerative clustering algorithm, k-means clustering, or K nearest neighbor algorithm on given problem instances. Judge if the method is appropriate for a given task.
Audience: Undergraduate

7. (Neural Networks and Deep Learning) Apply Perceptron learning rule on given problem instances. Implement neural networks using given software packages.
Audience: Undergraduate

8. (Reinforcement Learning) Understand the concepts of reinforcement learning. Identify and summarize its important features. Compute value function and Q function. Apply value iteration and Q learning on given problems.
Audience: Undergraduate

COMP SCI 541 – THEORY & ALGORITHMS FOR DATA SCIENCE
3 credits.

Theoretical methods for data science. Topics include: review of probability background, concentration inequalities, geometry of high dimensional random variables, parametric and non-parametric estimation, selected topics from optimization (optimality conditions; deterministic and stochastic gradient descent), PAC learning, sample complexity and algorithms for linear classification and regression, and property/distribution testing. Uses Python programming language.
Requisites: (COMP SCI 200, 220, placement into COMP SCI 300, or STAT 340), (MATH 320, 340, 341, 345, or 375), and (STAT 311, 333, 340, MATH/STAT 309, 431, MATH 331, 531, or I SY E 210), or graduate/professional standing
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Use and explain theoretical concepts in data science, such as concentration inequalities, learning, nonparametric distribution testing, optimization.
Audience: Undergraduate

2. Use properties of high-dimensional random variables to perform tasks such as dimension reduction and separating mixtures of Gaussians
Audience: Undergraduate

3. Use optimization algorithms such as gradient descent and stochastic gradient descent and analyze their computational complexity
Audience: Undergraduate

4. Analyze sample complexity for basic learning tasks such binary linear classification, as well as computational complexity of algorithms for solving them
Audience: Undergraduate

5. Analyze sample complexity of distribution testing tasks, such as testing uniformity and testing equivalence of discrete distributions
Audience: Undergraduate

COMP SCI 542 – INTRODUCTION TO SOFTWARE SECURITY
3 credits.

Teaches the security considerations that occur during all steps of the software development life cycle: methodologies for designing secure software, programming using secure programming techniques, in-depth vulnerability assessment methodologies, static and dynamic analysis tools for evaluating software security, and system defenses reducing security threats.
Requisites: COMP SCI 400 or 320, graduate/professional standing, or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Think like an attacker, that is to think about software the way that an adversary would.
Audience: Undergraduate

2. Design secure programs based on a structured methodology such as Threat Modeling. Have security in mind before writing the first line of code.

Audience: Undergraduate

3. Program in a secure way by mastering a comprehensive set of techniques for producing code that is more resilient to attack.
Audience: Undergraduate

4. Think like a security analyst, that is how to perform an in-depth software vulnerability assessments using a structured methodology such as First Principles Vulnerability Assessment (FPVA). Acquire the skills of a security analyst.
Audience: Undergraduate

5. Use a variety of automated tools that help statically and dynamically analyze code for security flaws.
Audience: Undergraduate

COMP SCI 544 – INTRODUCTION TO BIG DATA SYSTEMS
3 credits.

Deploy and use distributed systems to store and analyze large datasets. Unstructured and structured approaches to storage will be covered. Analysis will involve learning new query languages, processing streaming data, and training machine learning models. Most programming will be done in Python.
Requisites: COMP SCI 320, 400, or Graduate/Professional Standing
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Deploy distributed systems for data storage and analytics
Audience: Undergraduate

2. Demonstrate competencies with tools and processes necessary for loading data into distributed storage systems
Audience: Undergraduate

3. Write programs that use distributed platforms to efficiently analyze large datasets
Audience: Undergraduate

4. Produce meaning from large datasets by training machine learning models in parallel or on distributed systems
Audience: Undergraduate

5. Measure resource usage and overall cost of running distributed programs
Audience: Undergraduate

6. Optimize distributed analytics programs to reduce resource consumption and program runtime
Audience: Undergraduate

7. Demonstrate competencies with cloud services designed to store or analyze large datasets
Audience: Undergraduate

COMP SCI/E C E 552 – INTRODUCTION TO COMPUTER ARCHITECTURE
3 credits.

The design of computer systems and components. Processor design, instruction set design, and addressing; control structures and microprogramming; memory management, caches, and memory hierarchies; and interrupts and I/O structures. E C E 551 or knowledge of Verilog is recommended.
Requisites: (E C E/COMP SCI 352 and E C E/COMP SCI 354) or graduate/professional standing
Course Designation: Breadth - Physical Sci. Counts toward the Natural Sci req
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Use standard performance metrics to compare performance of different digital systems
Audience: Undergraduate

2. Design a pipelined data path for a RISC (reduced instruction set computer) instruction set and identify concepts of data dependence, pipelined hazards and out of order execution
Audience: Undergraduate

3. Design basic data and control cache subsystems and be able to operate basic memory systems
Audience: Undergraduate

4. Design a pipelined RISC micro-processor system with data cache using computer aided design tool and validate the correctness of the design using logic simulation
Audience: Undergraduate

COMP SCI 557 – PARALLEL & THROUGHPUT- OPTIMIZED PROGRAMMING
3 credits.

A comprehensive yet accessible introduction to high-performance computing practices, emphasizing shared-memory systems and using numerical algorithms as case studies. A focus on shared-memory, single-node multiprocessor systems, with some attention to GPUs. Emphasis is given on multithreaded programming, vectorization, memory hierarchy and its implications, and analysis of performance-limiting factors.
Requisites: E C E/COMP SCI 354 and (MATH 320, 340, 341, or 375), or graduate/professional standing
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Evaluate the efficiency and parallelism potential of algorithms, relative to bounds imposed by architectural limitations such as compute and memory bandwidth.
Audience: Undergraduate

2. Understand the implications of the memory system on parallel scaling, including considerations of memory throughput, latency and prefetching efficiency.
Audience: Undergraduate

3. Recognize instruction-level parallelism and evaluate how amenable different codes are to compiler-automated (or user-guided, via intrinsics) vectorization.
Audience: Undergraduate

4. Compare data layouts (dense vs. sparse data, structures-of-arrays vs. arrays-of-structures) to throughput-sensitive codes.
Audience: Undergraduate

5. Explore computational and memory-related performance implications of fundamental numerical codes such as convolutions, streaming vector operations, and matrix manipulations
Audience: Undergraduate

COMP SCI 559 – COMPUTER GRAPHICS
3 credits.

Survey of computer graphics. Image representation, formation, presentation, composition and manipulation. Modeling, transformation, and display of geometric objects in two and three dimensions. Representation of curves and surfaces. Rendering, animation, multi-media and visualization. Fluency with vector mathematics (e.g., from MATH 234 or a linear algebra class) is recommended.
Requisites: MATH 222 and (COMP SCI 367 or 400) or graduate/professional standing or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Develop interactive graphical applications in 2D and 3D using different types of APIs
Audience: Undergraduate

2. Analyze and apply coordinate systems and transformations in 2D and 3D to model objects hierarchically and prepare them for viewing
Audience: Undergraduate

3. Explain and apply shape modeling techniques, including parametric curves and meshes, in 2D and 3D
Audience: Undergraduate

4. Select and apply appearance modeling techniques including texturing and lighting
Audience: Undergraduate

5. Explain the structure and operation of the graphics pipeline and hardware, and use this understanding to implement and optimize efficient graphical applications
Audience: Undergraduate

6. Identify the issues in discrete representations and select approaches to mitigate them
Audience: Undergraduate

7. Discuss how imagery is perceived and the impact of perception on computer generated imagery
Audience: Undergraduate

COMP SCI/E C E 561 – PROBABILITY AND INFORMATION THEORY IN MACHINE LEARNING
3 credits.

Probabilistic tools for machine learning and analysis of real-world datasets. Introductory topics include classification, regression, probability theory, decision theory and quantifying information with entropy, relative entropy and mutual information. Additional topics include naive Bayes, probabilistic graphical models, discriminant analysis, logistic regression, expectation maximization, source coding and variational inference.
Requisites: (MATH 320, 340, 341, 375, or M E/COMP SCI/E C E 532 or concurrent enrollment) and (E C E 331, STAT/MATH 309, 431, STAT 311, 324, M E/STAT 424 or MATH 531) or grad/profsnl standing or declared in Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2024
Learning Outcomes: 1. Identify how ambiguity and noise leads to the need for probabilistic methods in machine learning
Audience: Both Grad & Undergrad

2. Implement classification, prediction and generative algorithms using a variety of techniques based in probability, information theory and machine learning
Audience: Both Grad & Undergrad

3. Prove optimality of a variety of algorithms and demonstrate understanding of sample complexity bounds
Audience: Graduate

COMP SCI 564 – DATABASE MANAGEMENT SYSTEMS: DESIGN AND IMPLEMENTATION
4 credits.

What a database management system is; different data models currently used to structure the logical view of the database: relational, hierarchical, and network. Hands-on experience with relational and network-based database systems. Implementation techniques for database systems. File organization, query processing, concurrency control, rollback and recovery, integrity and consistency, and view implementation.
Requisites: E C E/COMP SCI 354 and (COMP SCI 367 or 400) or graduate/professional standing or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Breadth - Physical Sci. Counts toward the Natural Sci req
Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Articulate and apply the foundations of relational database management systems (RDBMS).
Audience: Undergraduate

2. Explain the concepts of data models, including the Entity-Relationship (ER) and Relational models, express application in these models, and translate between models
Audience: Undergraduate

3. Use SQL to express complex queries over a relational database.
Audience: Undergraduate

4. Describe how an RDBMS stores, indexes, and accesses data.
Audience: Undergraduate

5. Explain how an RDBMS optimizes query execution and efficiently implements various operators.
Audience: Undergraduate

6. Explain the motivations for transactions, concurrency control, and crash recovery.
Audience: Undergraduate

7. Modify and extend the functionality of RDBMS.
Audience: Undergraduate

COMP SCI 565 – INTRODUCTION TO DATA VISUALIZATION
3 credits.

Introduction to topics such as perception, cognition, communication, design, implementation, applications, tools, and evaluation. Provides a broad survey of the field and covers fundamental concepts, theory, and tools in data visualization with opportunities for hands-on activities. Gain real-world experience in designing and evaluating visualizations.
Requisites: COMP SCI 320, 400, or Graduate/Professional Standing
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2024
Learning Outcomes: 1. Differentiate between visualizations to communicate within situational context.
Audience: Undergraduate

2. Illustrate the types of properties that make visualizations more effective.
Audience: Undergraduate

3. Apply techniques and algorithms for creating effective visualizations based on principles and techniques from graphic design, visual art, perceptual psychology, and cognitive science.
Audience: Undergraduate

4. Utilize visualization creation tools in individual and group assignment
Audience: Undergraduate

COMP SCI 566 – INTRODUCTION TO COMPUTER VISION
3 credits.

Topics include image formation, feature detection, motion estimation, image mosaics, 3D shape reconstruction, and object recognition. Applications of these techniques include building 3D maps, creating virtual characters, organizing photo and video databases, human computer interaction, video surveillance, and automatic vehicle navigation. Broad overview of various computer vision and machine learning techniques and sensing and imaging technologies used in computer vision applications. Project-based.
Requisites: COMP SCI 400 and (MATH 320, 340, 341, 345 or 375) and (STAT 311, 324, 333, 340, 371, STAT/MATH 309, 431, MATH 331 or 531) or graduate/professional standing
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Describe a broad range of fundamental concepts in 2D and 3D computer vision, including basic image processing, 2D image recognition, 3D sensing and motion recovery
Audience: Undergraduate

2. Explain basic theories, current approaches, key concepts, and common practices in computer vision.
Audience: Undergraduate

3. Design basic statistics and machine learning algorithms for computer vision problems, conduct experiments to evaluate the proposed approaches, and analyze and interpret the results.
Audience: Undergraduate

4. Develop basic computer vision applications and projects in MATLAB.
Audience: Undergraduate

5. Prepare and deliver clear and concise oral presentations.
Audience: Undergraduate

COMP SCI/B M I 567 – BIOMEDICAL IMAGE ANALYSIS
3 credits.

Hands-on introduction to biological and medical image analysis techniques. Topics include medical imaging formats, segmentation, registration, image quantification, and classification.
Requisites: (MATH 320, 340, 341, 345, or 375) and (STAT 511, 541, POP HLTH/B M I 551, MATH 331, MATH/STAT 431, 309, STAT 240, 301, 311, 324, 371, or STAT/F&W ECOL 571) or graduate/professional standing
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Implement the key principles of ideas from probability, statistics and computer vision algorithms used in medical image analysis
Audience: Undergraduate

2. Recognize which image analysis problems will benefit from which modeling approach
Audience: Undergraduate

3. Apply algorithms about image analysis tasks and implement algorithms and pipelines using a programming language
Audience: Undergraduate

4. Implement the building blocks taught in this course to independently learn and apply new, but related imaging analysis algorithms
Audience: Undergraduate

COMP SCI 570 – INTRODUCTION TO HUMAN-COMPUTER INTERACTION
3 credits.

User-centered software design: (1) principles of and methods for understanding user needs; designing and prototyping interface solutions; and evaluating their usability, (2) their applications in designing multiple types of interfaces through group projects.
Requisites: Junior standing or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Understand fundamental concepts and principles of Human-Computer Interaction (HCI).
Audience: Undergraduate

2. Apply user-centered design principles and methods to create effective and usable interfaces.
Audience: Undergraduate

3. Analyze and evaluate user interfaces using appropriate usability evaluation techniques.
Audience: Undergraduate

4. Develop a toolbox of user-centered design techniques that can be applied to new design problems.
Audience: Undergraduate

5. Understand user needs and expectations with the aid of user research methods.
Audience: Undergraduate

COMP SCI 571 – BUILDING USER INTERFACES
3 credits.

Introduces software development of user interfaces (UIs). Build competence in implementing UIs using state-of-the-art (1) UI paradigms, such as event-driven interfaces, direct-manipulation interfaces, and dialogue-based interaction; (2) methods for capturing, interpreting, and responding to different forms of user input and states, including pointing, text entry, speech, touch, gestures, user activity, context, and physiological states; and (3) platform-specific UI development APIs, frameworks, and toolkits for multiple platforms including web/mobile/desktop interfaces, natural user interfaces, and voice user interfaces. Learn about the fundamental concepts, technologies, algorithms, and methods in building user interfaces, implement UIs using of state-of-the-art UI development tools, and build a UI development portfolio.
Requisites: COMP SCI 400
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Engage in design thinking around user interface needs and problems, ideate and communicate conceptual design solutions

Audience: Undergraduate

2. Create visual designs, layouts, and navigation structures, and effectively use design languages, color palettes, and platform-specific design elements
Audience: Undergraduate

3. Prototype and develop user interfaces for the Web, mobile, and voice user interfaces (VUIs)
Audience: Undergraduate

4. Program front-end, user-facing software elements using the state-of-the-art programming languages, frameworks, and libraries
Audience: Undergraduate

5. Follow user-centered design principles, heuristics, and methods to iteratively build, assess, and refine design solutions
Audience: Undergraduate

COMP SCI/B M I 576 – INTRODUCTION TO BIOINFORMATICS
3 credits.

Algorithms for computational problems in molecular biology. Studies algorithms for problems such as: genome sequencing and mapping, pairwise and multiple sequence alignment, modeling sequence classes and features, phylogenetic tree construction, and gene-expression data analysis.
Requisites: (COMP SCI 320 or 400) and MATH 222, graduate/professional standing, or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Explain the biology and significance of the most commonly measured molecules in molecular biology.
Audience: Undergraduate

2. Identify the primary computational problems associated with each type of biological data.
Audience: Undergraduate

3. Explain the major algorithms and approaches used to address the computational problems.
Audience: Undergraduate

4. Implement efficient algorithms for bioinformatic tasks through the use of the discussed approaches.
Audience: Undergraduate

5. Apply the discussed algorithms to novel but closely-related tasks.
Audience: Undergraduate

6. Understand the methods covered such that parts of the methods sections of published biological papers are interpretable.
Audience: Undergraduate

7. Begin to gain the qualifications of a bioinformatician.
Audience: Undergraduate

COMP SCI 577 – INTRODUCTION TO ALGORITHMS
4 credits.

Basic paradigms for the design and analysis of efficient algorithms: greed, divide-and-conquer, dynamic programming, reductions, and the use of randomness. Computational intractability including typical NP-complete problems and ways to deal with them.
Requisites: (MATH/COMP SCI 240 or STAT/COMP SCI/MATH 475) and (COMP SCI 367 or 400), or graduate/professional standing, or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Design and analyze efficient algorithms based on the paradigms of divide-and-conquer, dynamic programming, and greed.
Audience: Undergraduate

2. Construct clear, concise, and sound proofs for algorithmic correctness and complexity statements, applying logical reasoning, synthesizing definitions and theorems, and effectively communicating structured arguments.
Audience: Undergraduate

3. Formulate abstractions of computational problems, and design and analyze efficient reductions between computational problems.
Audience: Undergraduate

4. Describe, select and apply paradigmatic algorithms and reductions dealing with numbers, strings, graphs, and networks, including some that go beyond the standard paradigms and/or involve more intricate mathematics.
Audience: Undergraduate

5. Recognize computational intractability, demonstrate NP-hardness, and articulate the repercussions.
Audience: Undergraduate

COMP SCI 578 – CONTEST-LEVEL PROGRAMMING
1 credit.

Training in computer programming for competitions: assessing the coding difficulty and complexity of computational problems, recognizing the applicability of known algorithms, fast coding and testing, team work. COMP SCI 577 is suggested but not required.
Requisites: (COMP SCI 300 or 367), graduate/professional standing, or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Fall 2020

COMP SCI/DS 579 – VIRTUAL REALITY
3 credits.

Introduces the field of virtual reality with an emphasis on creating immersive, interactive virtual experiences. Survey topics include the history of virtual reality technology, computer graphics and 3D modeling, human perception and psychology, and principles of human-computer interaction and user interface design. Is intended for individuals with backgrounds in Computer Science, Engineering, Art, Architecture, or Design.
Requisites: Sophomore standing
Course Designation: Level - Intermediate
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Define and describe concepts related to virtual reality.
Audience: Undergraduate

2. Demonstrate a basic understanding of how interactive computer graphics and game engines are built and operate.
Audience: Undergraduate

3. Identify the underlying components of virtual reality software and hardware.
Audience: Undergraduate

4. Identify the potential applications and challenges of virtual reality.
Audience: Undergraduate

5. Apply principles of virtual reality in the construction of a class project.
Audience: Undergraduate

COMP SCI/L I S 611 – USER EXPERIENCE DESIGN 1
3 credits.

Introduction to the user experience design including key stages of the design process, design ethics, and the methods and tools involved at each stage of design. Conduct formative research on clients, users, use contexts and tasks. Gain experience with user research methodologies and learn to create intermediate design tools such as personas. Develop and present a design proposal for a chosen project.
Requisites: Declared in Information MS, Design + Innovation MS , or Capstone Certificate in User Experience Design
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Apply fundamental concepts and practices of user experience design
Audience: Graduate

2. Understand the ethics of design including practices of inclusive design and accessibility
Audience: Graduate

3. Conduct formative research to inform design
Audience: Graduate

4. Apply common user data collection methods
Audience: Graduate

5. Analyze and visualize processes across time and interfaces
Audience: Graduate

6. Create and apply common UX design tools such as personas, scenarios and user journey maps
Audience: Graduate

7. Effectively convey the output of user research and initial design through oral and written communication.
Audience: Graduate

COMP SCI/L I S 612 – USER EXPERIENCE DESIGN 2
3 credits.

Advanced study of UX design. Introduces processes of ideation, key concepts of visual design, conceptual and interaction design, low and high-resolution prototyping of design techniques. Applications include drafting designs based on user models and initial testing of prototypes.
Requisites: COMP SCI/L I S 611 and Declared in Information MS, Design + Innovation MS, or Capstone Certificate in User Experience Design
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Develop design ideas and communicate them through brainstorming, sketching, and modeling;
Audience: Graduate

2. Create designs that follow principles of and best practices in visual and interaction design;
Audience: Graduate

3. Prototype designs using rapid prototyping methods for communication and testing;
Audience: Graduate

4. Understand human perceptual, cognitive, and motor processes involved in interaction;
Audience: Graduate

5. Evaluate designs using expert- and empirical-evaluation methods;
Audience: Graduate

6. Integrate design, prototyping, and evaluation methods and principles into a process toward addressing a design problem
Audience: Graduate

7. Communicate their ideas to others, integrate feedback into their design work, and critique the work of others constructively.
Audience: Graduate

COMP SCI/L I S 613 – USER EXPERIENCE DESIGN 3
3 credits.

Conduct formal evaluations of the user experience (UX) or usability of a digital system. Gain familiarity with the evaluation and research process including key stages, tasks for each stage, common data collection and analysis methods, and common tools employed in the field. Gain experience with a variety of UX evaluation approaches. Collect pilot data and develop a proposal for further UX testing.
Requisites: COMP SCI/L I S 612 and Declared in Information MS, Design + Innovation MS, or Capstone Certificate in User Experience Design
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Demonstrate understanding of a variety of UX testing approaches
Audience: Graduate

2. Plan and implement all phases of testing for a digital system including planning, data collection, analysis and reporting
Audience: Graduate

3. Demonstrate understanding of the relationship among research design, instruments, metrics, and data analysis
Audience: Graduate

4. Implement major testing approaches such as task-based, information architecture and accessibility
Audience: Graduate

5. Have knowledge of contemporary tools used for UX testing
Audience: Graduate

6. Communicate evaluation findings effectively and use data to improve systems design
Audience: Graduate

COMP SCI/L I S 614 – USER EXPERIENCE DESIGN CAPSTONE
1 credit.

Applies a design studio critique approach to produce a learning environment of collaborative and interdisciplinary peer critique and learning, in addition to provide expert feedback and suggestions. Present and defend the latest iteration of the user experience design project developed in earlier courses while learning about the professions associated with digital user experience design.
Requisites: COMP SCI/L I S 613 and declared in Design + Innovation MS, or the Capstone Certificate in User Experience Design
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Summer 2025
Learning Outcomes: 1. Knowledge of, and ability to apply, data collection and analysis methodologies for user experience research.
Audience: Graduate

2. Knowledge of, and ability to apply, design principles and user behavior theories to digital environments.
Audience: Graduate

3. Create, critique and revise design prototypes based on testing data and feedback
Audience: Graduate

4. Effectively plan, manage and communicate a user experience design project.
Audience: Graduate

COMP SCI 620 — COMPUTER SCIENCES CAPSTONE
3 credits.

Build a meaningful product from start to finish with a local, regional, national and international corporate client that solves a real-world problem. In a collaborative space design, develop, test, debug, document, and deliver a software project for a corporate client, learning and using new technologies and agile software development techniques.
Requisites: COMP SCI 400, senior standing, and declared in an undergraduate Computer Sciences major
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Evaluate and utilize agile techniques such as Design Thinking and Scrum.
Audience: Undergraduate

2. Demonstrate effective teamwork skills.
Audience: Undergraduate

3. Develop a software product for a corporate client from start to finish including phases for discovery, design, coding, testing, and documentation.
Audience: Undergraduate

4. Execute presentation skills by holding weekly demos with corporate client.
Audience: Undergraduate

5. Examine the types of roles typically present in an agile software development organization such as Product Manager, UX, Scrum Master, engineers, testers, DevOps, etc
Audience: Undergraduate

COMP SCI 638 — UNDERGRADUATE TOPICS IN COMPUTING
1-4 credits.

Selected topics in computing. Each offering of the course will cover a topic selected by the instructor and may cover one or more topics from all of computer science.
Requisites: COMP SCI 200, 300, 301, 302, 310, 367, placement into COMP SCI 300, or L I S/COMP SCI 102 (COMP SCI 202 prior to Fall 2023), graduate/professional standing, or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Breadth - Natural Science
Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2025

COMP SCI 639 — UNDERGRADUATE ELECTIVE TOPICS IN COMPUTING
3-4 credits.

Selected topics in computing. Each offering of the course will cover a topic selected by the instructor. Offerings of this course will provide sufficient depth into their subject to count as electives to meet CS Major requirements.
Requisites: None
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 640 — INTRODUCTION TO COMPUTER NETWORKS
3 credits.

Architecture of computer networks and network protocols, protocol layering, reliable transmission, congestion control, flow control, naming and addressing, unicast and multicast routing, network security, network performance widely used protocols such as Ethernet, wireless LANs, IP, TCP, and HTTP.
Requisites: (COMP SCI/E C E 354 and COMP SCI 400) or graduate/professional standing
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Understand what a computer network is and its key components, such as switches, routers, firewalls, and access points.
Audience: Undergraduate

2. Explain the concept of network protocols and their role in communications, how the Internet works, and especially the OSI model and the TCP/IP protocol suite.
Audience: Undergraduate

3. Analyze network protocols at different layers, including TCP, UDP, IP, ARP, the IEEE 802 suite, DNS, BGP, and more, and including how they interact with each other.
Audience: Undergraduate

4. Evaluate network performance metrics, strategies that allow for effective navigation and troubleshooting of network issues, and approaches to design network architectures.
Audience: Undergraduate

5. Design and implement network services and applications.
Audience: Undergraduate

COMP SCI 642 – INTRODUCTION TO INFORMATION SECURITY
3 credits.

Senior level undergraduate course covering various topics on information security. Covers a wide range of topics, such as cryptographic primitives, security protocols, system security, and emerging topics. Elementary knowledge of mathematical logic and discrete probability theory needed, such as MATH/COMP SCI 240.
Requisites: COMP SCI 537 or graduate/professional standing or declared in the Capstone Certificate in Computer Sciences for Professionals
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Explain the purpose of security and accurately define and apply key concepts, including threats, vulnerabilities, and trust.
Audience: Undergraduate

2. Apply best practices for both operational and system design security.
Audience: Undergraduate

3. Apply formal security properties provided by cryptographic techniques and illustrate them with relevant cryptographic algorithms.
Audience: Undergraduate

4. Evaluate the three fundamental authentication mechanisms, evaluate their trade-offs, and determine appropriate use cases.
Audience: Undergraduate

5. Define, implement, and enforce access control policies within networks and operating systems.
Audience: Undergraduate

6. Analyze the security of network and web environments through the deployment technology and policy
Audience: Undergraduate

7. Create threat and trust models, and apply appropriate security mechanisms to protect system integrity.
Audience: Undergraduate

COMP SCI 681 – SENIOR HONORS THESIS
3 credits.

Individual study for seniors completing theses for honors in the Computer Sciences major as arranged with a faculty member.
Requisites: Consent of instructor
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Honors - Honors Only Courses (H)
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 682 – SENIOR HONORS THESIS
3 credits.

Individual study for seniors completing theses for honors in the Computer Sciences major as arranged with a faculty member. Continuation of COMP SCI 681
Requisites: Consent of instructor
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Honors - Honors Only Courses (H)
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 691 – SENIOR THESIS
2-3 credits.

Individual study for seniors completing theses as arranged with a faculty member.
Requisites: Consent of instructor
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2025

COMP SCI 692 – SENIOR THESIS
2-3 credits.

Individual study for seniors completing theses as arranged with a faculty member, continuation of COMP SCI 691
Requisites: Consent of instructor
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Fall 2025

COMP SCI 698 – DIRECTED STUDY
1-6 credits.

Directed study projects for juniors and seniors as arranged with a faculty member.
Requisites: Consent of instructor
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Fall 2025

COMP SCI 699 – DIRECTED STUDY
1-6 credits.

Directed study projects for juniors and seniors as arranged with a faculty member.
Requisites: Consent of instructor
Course Designation: Level - Advanced
L&S Credit - Counts as Liberal Arts and Science credit in L&S
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 701 – CONSTRUCTION OF COMPILERS
3 credits.

Principles of the design and implementation of programming languages. Topics include: Principles of compilation, static program analysis, compilation methods to support profiling, and code-generation methods. Knowledge of programming languages and compiler design strongly encouraged, such as COMP SCI 536.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2020

COMP SCI 702 – GRADUATE COOPERATIVE EDUCATION
1-2 credits.

Apply computer sciences principles in a hands-on, professional setting.
Requisites: Consent of instructor
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026
Learning Outcomes: 1. Understand the nature and demands of a professional career in computer science.
Audience: Graduate

2. Apply knowledge gained in computer science coursework in a professional setting.
Audience: Graduate

COMP SCI 703 – PROGRAM VERIFICATION AND SYNTHESIS
3 credits.

Formal methods for program verification. Model-checking techniques; linear temporal logic; computational tree logic; logic/automata connection; bisimulations; probabilistic model-checking. Special topics include: program synthesis, verification and synthesis of privacy properties. Knowledge of programming languages and compiler design strongly encouraged, such as COMP SCI 536.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2025
Learning Outcomes: 1. Logic: Syntax and formal semantics of temporal logics such as LTL and MSO and formalizing software safety properties. Translations from temporal logics to automata.
Audience: Graduate

2. Model Checking: Understand algorithms for checking whether a program satisfies a temporal property in temporal logic and apply such algorithms to verification problems.
Audience: Graduate

3. Synthesis: Understand and apply state-of-the-art program synthesis algorithms such as counterexample-guided inductive synthesis, syntax-guided synthesis, sketching, and MCMC sampling.
Audience: Graduate

COMP SCI 704 – PRINCIPLES OF PROGRAMMING LANGUAGES
3 credits.

Introduction to principles of advanced programming languages and programming-language theory. Topics include: lambda-calculus, functional languages, polymorphic functions, type inference, structural induction, lazy evaluation, operational semantics, denotational semantics, and axiomatic semantics. Students are strongly encouraged to have knowledge of programming languages, such as from COMP SCI 536.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI/E C E 707 – MOBILE AND WIRELESS NETWORKING
3 credits.

Design and implementation of protocols, systems, and applications for mobile and wireless networking, particularly at the media access control, network, transport, and application layers. Focus is on the unique problems and challenges presented by the properties of wireless transmission, various device constraints such as limited battery power, and node mobility. Knower of computer networking is strongly encouraged, such as from COMP SCI 640 or E C E 537.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI 710 – COMPUTATIONAL COMPLEXITY
3 credits.

Study of the capabilities and limitations of efficient computation. Relationships between models representing capabilities such as parallelism, randomness, quantum effects, and non-uniformity; and models based on the notions of nondeterminism, alternation, and counting, which capture the complexity of important problems. Knowledge of the theory of computation is strongly encouraged, such as COMP SCI 520.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI/MATH 714 – METHODS OF COMPUTATIONAL MATHEMATICS I
3 credits.

Development of finite difference methods for hyperbolic, parabolic and elliptic partial differential equations. Analysis of accuracy and stability of difference schemes. Direct and iterative methods for solving linear systems. Introduction to finite volume methods. Applications from science and engineering. Students are strongly encouraged to have programming skills (e.g. COMP SCI 200) and some undergraduate numerical analysis (e.g. MATH/COMP SCI 514 or COMP SCI 412), analysis and differential equations (e.g. MATH 322 and MATH 521) and linear algebra (e.g. MATH 341).
Requisites: Graduate/professional standing or member of the Pre-Masters Mathematics (Visiting International) Program
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI/MATH 715 – METHODS OF COMPUTATIONAL MATHEMATICS II
3 credits.

Introduction to spectral methods (Fourier, Chebyshev, Fast Fourier Transform), finite element methods (Galerkin methods, energy estimates and error analysis), and mesh-free methods (Monte-Carlo, smoothed-particle hydrodynamics) for solving partial differential equations. Applications from science and engineering. Applications from science and engineering. Students are strongly encouraged to have programming skills (e.g. COMP SCI 200), undergraduate numerical analysis (e.g. MATH/COMP SCI 514 or COMP SCI 412), analysis (MATH 322 and 521) and linear algebra (e.g. MATH 341 or equiv.)
Requisites: Graduate/professional standing or member of the Pre-Masters Mathematics (Visiting International) Program
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026

COMP SCI/I SY E 719 – STOCHASTIC PROGRAMMING
3 credits.

Stochastic programming is concerned with decision making in the presence of uncertainty, where the eventual outcome depends on a future random event. Topics include modeling uncertainty in optimization problems, risk measures, stochastic programming algorithms, approximation and sampling methods, and applications. Students are strongly encouraged to have knowledge of linear programming (e.g., MATH/COMP SCI/I SY E/STAT 525) and probability and statistics (e.g., MATH/STAT 431). Knowledge of integer optimization (MATH/COMP SCI/I SY E 728) is helpful, but not required.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2025
Learning Outcomes: 1. Learn the terms, basic capabilities, and limitations of stochastic programming models
Audience: Graduate

2. Formulate stochastic programming models
Audience: Graduate

3. Implement the algorithms used to solve stochastic programming problems
Audience: Graduate

4. Learn principles of decomposition algorithms for solving large-scale optimization problems
Audience: Graduate

COMP SCI/B M I/E C E/MED PHYS 722 – COMPUTATIONAL OPTICS AND IMAGING
3 credits.

Computational imaging includes all imaging methods that produce images as a result of computation on collected signals. Learn the tools to design new computational imaging methods to solve specific imaging problems. Provides an understanding of the physics of light propagation and measurement, and the computational tools to model it, including wave propagation, ray tracing, the radon transform, and linear algebra using matrix and integral operators and the computational tools to reconstruct an image, including linear inverse problems, neural networks, convex optimization, and filtered back-projection. Covers a variety of example computational imaging techniques and their applications including coded apertures, structured illumination, digital holography, computed tomography, imaging through scattering media, compressed sensing, and non-line-of-sight imaging.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2024
Learning Outcomes: 1. Apply ray and wave based light propagation models
Audience: Graduate

2. Explain the process of image formation in conventional imaging systems using theory and computational models
Audience: Graduate

3. Select and combine the different components required in an imaging system to perform light manipulation, collection, and image reconstruction

Audience: Graduate

4. Apply the linear matrix and integral operators that model light propagation
Audience: Graduate

5. Apply the linear inverse algorithms that allow an imaging system to reconstruct properties of the scene from collected data
Audience: Graduate

6. Simulate different computational imaging systems and perform computation on simulated datasets
Audience: Graduate

7. Understand the most common computational imaging techniques and be able to use and adapt them for their own applications
Audience: Graduate

COMP SCI/I SY E 723 – DYNAMIC PROGRAMMING AND ASSOCIATED TOPICS
3 credits.

General and special techniques of dynamic programming developed by means of examples. Shortest-path algorithms. Deterministic equipment replacement models. Resource allocation problem. Traveling-salesman problem. Knapsack problem. Analysis of inventory systems. General stochastic formulations. Markovian decision processes. Students are strongly encouraged to have knowledge of mathematical optimization (e.g., COMP SCI/I SY E/MATH/STAT 525, I SY E 623, COMP SCI/I SY E/MATH/STAT 726), knowledge of analysis (e.g., MATH/STAT 431 or 521) and programming ability (e.g., COMP SCI 200 or 301)
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Identify basic components, such as the state space, of a dynamic program
Audience: Graduate

2. Formulate and solve dynamic programs under different performance criteria such as finite horizon discounted reward/cost criteria
Audience: Graduate

3. Apply dynamic programming tools and concepts in `traditional' industrial engineering applications such as supply chain, manufacturing, and healthcare
Audience: Graduate

COMP SCI/I SY E/MATH/STAT 726 – NONLINEAR OPTIMIZATION I
3 credits.

Theory and algorithms for nonlinear optimization, focusing on unconstrained optimization. Line-search and trust-region methods; quasi-Newton methods; conjugate-gradient and limited-memory methods for large-scale problems; derivative-free optimization; algorithms for least-squares problems and nonlinear equations; gradient projection algorithms for bound-constrained problems; and simple penalty methods for nonlinearly constrained optimization. Students are strongly encouraged to have knowledge of linear algebra and familiarity with basic mathematical analysis.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026

COMP SCI/I SY E 727 — CONVEX ANALYSIS
3 credits.

Convex sets in finite-dimensional spaces: relative interiors, separation, set operations. Convex functions: conjugacy, subdifferentials and directional derivations, functional operations, Fenchel-Rockafellar duality. Applications to operations research and related areas. Students taking this course are strongly encouraged to have had a course in basic analysis (e.g. MATH 521) and a course in linear algebra (e.g., MATH 340).
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2025

COMP SCI/I SY E/MATH 728 — INTEGER OPTIMIZATION
3 credits.

Introduces optimization problems over integers, and surveys the theory behind the algorithms used in state-of-the-art methods for solving such problems. Special attention is given to the polyhedral formulations of these problems, and to their algebraic and geometric properties. Applicability of Integer Optimization is highlighted with applications in combinatorial optimization. Key topics include: formulations, relaxations, polyhedral theory, cutting planes, decomposition, enumeration. Students are strongly encouraged to have knowledge of Linear Programming (e.g., MATH/COMP SCI/I SY E/STAT 525), including algorithms, duality and polyhedral theory.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Describe and explain the basics of polyhedral theory, which consists in the study of systems of linear inequalities both from an algebraic and a geometric point of view
Audience: Graduate

2. Define perfect formulations and identify what properties are desirable in an integer programming formulation of a problem
Audience: Graduate

3. Explain how valid inequalities can be used as cutting planes to strengthen integer programming formulations
Audience: Graduate

COMP SCI/I SY E/MATH 730 — NONLINEAR OPTIMIZATION II
3 credits.

Theory and algorithms for nonlinearly constrained optimization. Relevant geometric concepts, including tangent and normal cones, theorems of the alternative, and separation results. Constraint qualifications. Geometric and algebraic expression of first-order optimality conditions. Second-order optimality conditions. Duality. Nonlinear programming algorithms: merit functions and filters; interior-point, augmented Lagrangian, and sequential quadratic programming algorithms.
Requisites: STAT/COMP SCI/I SY E/MATH 726
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026

COMP SCI 736 — ADVANCED OPERATING SYSTEMS
3 credits.

Advanced topics in operating systems, including process communication, resource allocation, multiprocess and network operating systems, kernel philosophies, fault-tolerant systems, virtual machines, high-level language systems, verifiability and proof techniques. Comp Sci 537 or cons inst
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI 739 — DISTRIBUTED SYSTEMS
3 credits.

Large-scale distributed systems have become pervasive, underlying virtually all widely used computing services. Explore prevalent issues in designing and implementing distributed systems. Topics include fault tolerance, scalability, replication, distributed storage, consensus, reliability, performance, and correctness.
Requisites: COMP SCI 736, 744, 764, or 774
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Describe core concepts in distributed computing, such as logical clocks, consensus, partitioning, replication and fault tolerance, etc.
Audience: Graduate

2. Explain how real-world distributed systems are realized by relying on core distributed computing concepts.
Audience: Graduate

3. Analyze how emerging technology trends concerning new hardware and applications, influence the designs and implementations of distributed systems.
Audience: Graduate

4. Solve and develop distributed computing problems, services and applications.
Audience: Graduate

COMP SCI 740 — ADVANCED COMPUTER NETWORKS
3 credits.

Advanced topics in computer communications networks: congestion and flow control; routing; rate-based protocols; high speed interfaces and technologies: metropolitan area networks; fast packet switching technologies; advanced applications; network services: name service, authentication, resource location. Students are strongly encouraged to have knowledge of computer network design and protocols (e.g., COMP SCI 640)
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI 744 – BIG DATA SYSTEMS
3 credits.

Issues in the design and implementation of big data processing systems, including: an overview of cluster architecture, key design goals (flexibility, performance and fault tolerance), popular execution frameworks, basic abstractions, and applications (e.g., batch analytics, stream processing, graph processing, and machine learning).
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2025

COMP SCI/E C E 750 – REAL-TIME COMPUTING SYSTEMS
3 credits.

Introduction to the unique issues in the design and analysis of computer systems for real-time applications. Hardware and software support for guaranteeing timeliness with and without failures. Resource management, time-constrained communication, scheduling and imprecise computations, real-time kernels and case studies. Students are strongly encouraged to have knowledge of computer architecture (e.g., E C E/COMP SCI 552) and operating system functions (e.g., COMP SCI 537)
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2024

COMP SCI/E C E 752 – ADVANCED COMPUTER ARCHITECTURE I
3 credits.

Processor design, computer arithmetic, pipelining, multi-operation processors, vector processors, control units, precise interrupts, main memory, cache memories, instruction set design, stack machines, busses and I/O, protection and security. Students are strongly encouraged to have knowledge of computer architecture (e.g., E C E/COMP SCI 552).
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI/E C E 755 – VLSI SYSTEMS DESIGN
3 credits.

Overview of MOS devices and circuits; introduction to integrated circuit fabrication; topological design of data flow and control; interactive graphics layout; circuit simulation; system timing; organizational and architectural considerations; alternative implementation approaches; design project. E C E 555 or equivalent experience is strongly recommended.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026

COMP SCI/E C E 756 – COMPUTER-AIDED DESIGN FOR VLSI
3 credits.

Broad introduction to computer-aided design tools for VLSI, emphasizing implementation algorithms and data structures. Topics covered: design styles, layout editors, symbolic compaction, module generators, placement and routing, automatic synthesis, design-rule checking, circuit extraction, simulation and verification. Students are strongly encourage to have programming skills and to have taken a course in Digital System Fundamentals such as E C E/COMP SCI 352.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2023

COMP SCI/E C E 757 – ADVANCED COMPUTER ARCHITECTURE II
3 credits.

Parallel algorithms, principles of parallelism detection and vectorizing compilers, interconnection networks, MIMD machines, processor synchronization, data coherence, multis, dataflow machines, special purpose processors. Students are strongly encouraged to have knowledge of computer architecture (e.g., E C E/COMP SCI 552).
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026

COMP SCI 758 – ADVANCED TOPICS IN COMPUTER ARCHITECTURE
3 credits.

Advanced topics in computer architecture that explore the implications to architecture of forthcoming evolutionary and revolutionary changes in application demands, software paradigms, and hardware implementation technologies. Students are strongly encouraged to have knowledge of computer architecture (e.g., E C E/COMP SCI 552).
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI/E C E/E M A/E P/M E 759 – HIGH PERFORMANCE COMPUTING FOR APPLICATIONS IN ENGINEERING
3 credits.

An overview of hardware and software solutions that enable the use of advanced computing in tackling computationally intensive Engineering problems. Hands-on learning promoted through programming assignments that leverage emerging hardware architectures and use parallel computing programming languages. Students are strongly encourage to have completed COMP SCI 367 or COMP SCI 400 or to have equivalent experience.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026

COMP SCI/E C E 760 – MACHINE LEARNING
3 credits.

Computational approaches to learning: including inductive inference, explanation-based learning, analogical learning, connectionism, and formal models. What it means to learn. Algorithms for learning. Comparison and evaluation of learning algorithms. Cognitive modeling and relevant psychological results.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Identify different aspects of machine learning, including supervised learning, unsupervised learning, and reinforcement learning
Audience: Graduate

2. Implement and analyze a variety of supervised models for classification and regression, including decision trees, instance-based models, naive Bayes, support vector machines, a variety of neural networks, linear and logistic regression, and others
Audience: Graduate

3. Implement and analyze neural network models, starting with the perceptron, and continuing to multilayer perceptrons, convolutional neural networks, recurrent neural networks, along with deep generative models
Audience: Graduate

4. Identify various types of regularization techniques and their properties
Audience: Graduate

5. Implement optimization techniques used in modern machine learning, including gradient descent and stochastic gradient descent
Audience: Graduate

6. Apply various concepts and metrics involved in evaluating models: accuracy, F measures, ROC, and precision/recall curves, and implement cross-validation
Audience: Graduate

7. Analyze unsupervised learning techniques for clustering, dimensionality reduction, and latent models
Audience: Graduate

8. Identify classical and modern techniques to improve models or deal with dearth of data: ensemble methods, semi-supervised learning, weak supervision
Audience: Graduate

COMP SCI/E C E 761 – MATHEMATICAL FOUNDATIONS OF MACHINE LEARNING
3 credits.

Mathematical foundations of machine learning theory and algorithms. Probabilistic, algebraic, and geometric models and representations of data, mathematical analysis of state-of-the-art learning algorithms and optimization methods, and applications of machine learning. Knowledge of probability [such as MATH/STAT 431 or COMP SCI/E C E 561] and linear algebra [such as MATH 341 or M E/COMP SCI/E C E 532] is required.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Derive and apply mathematical tools for machine learning from probability, statistics, linear algebra, and optimization
Audience: Graduate

2. Perform mathematical analysis and characterization of generative and discriminative models
Audience: Graduate

3. Perform mathematical analysis of machine learning algorithms
Audience: Graduate

4. Perform derivation of basic machine learning error bounds and related performance analysis
Audience: Graduate

5. Read and understand theoretical papers from machine learning conferences
Audience: Graduate

COMP SCI 762 – ADVANCED DEEP LEARNING
3 credits.

Explore methods and applications of deep learning. Covers cutting-edge topics, including neural architecture design, robustness and reliability of deep learning, learning with less supervision, lifelong machine learning, deep generative modeling, theoretical understanding of deep learning, and interpretable deep learning.
Requisites: E C E/COMP SCI 760
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Understand the key elements and methods in the design and use of deep neural networks;
Audience: Graduate

2. Advance knowledge and get exposure to cutting-edge topics in AI and deep learning
Audience: Graduate

3. Explore new research directions and applications of deep learning
Audience: Graduate

4. Identify and participate in original research in a collaborative team
Audience: Graduate

5. Search for sources of information and evaluation methods and tools relevant to the project
Audience: Graduate

6. Get hands-on experience in writing software and programs using popular deep learning libraries and frameworks
Audience: Graduate

7. Develop analytic and problem-solving skills using computational approaches
Audience: Graduate

COMP SCI/E C E 763 – TRUSTWORTHY ARTIFICIAL INTELLIGENCE
3 credits.

Explore security and privacy aspects of trustworthy artificial intelligence. Three core subjects will be considered: differential privacy and algorithmic fairness; adversarial machine learning; and end-to-end trustworthy systems. A selection of more advanced topics may be covered such as additional notions of privacy, language-based security, and robust optimization. Knowledge of probability/statistics (such as MATH 431), cryptography (such as MATH 435), security (such as COMP SCI 642), and modern machine learning (such as M E/COMP SCI/E C E 539 or 540) is required.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Evaluate machine learning and AI systems from an adversarial, security and privacy mindset.
Audience: Graduate

2. Identify common pitfalls and problems in ensuring security and privacy for AI.
Audience: Graduate

3. Summarize the commonalities and differences between notions of security and privacy (e.g., the difference between privacy and cryptographic security).
Audience: Graduate

4. Explain the strengths and limitations of candidate definitions of robustness, security, privacy, and fairness properties in AI.
Audience: Graduate

5. Apply useful primitives from end-to-end trustworthiness to machine learning and AI systems.
Audience: Graduate

6. Use modern tools to design attacks and implement defensive measures.
Audience: Graduate

COMP SCI 764 – TOPICS IN DATABASE MANAGEMENT SYSTEMS
3 credits.

Implementation of database management systems, the impact of new technology on database management systems, back-end database computers, distributed database management systems, concurrency control, and query execution in both distributed and centralized systems, implementation of multiple user views, roll-back and recovery mechanisms, database translation. Students are strongly encouraged to have knowledge of database design (e.g., COMP SCI 564).
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI 765 – DATA VISUALIZATION
3 credits.

Principles of the visual presentation of data. Survey of Information Visualization, Scientific Visualization, and Visual Analytics. Design and evaluation of visualizations and interactive exploration tools. Introduction to relevant foundations in visual design, human perception, and data analysis. Encodings, layout and interaction. Approaches to large data sets. Visualization of complex data types such as scalar fields, graphs, sets, texts, and multi-variate data. Use of 2D, 3D and motion in data presentations. Implementation issues.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI/E C E 766 – COMPUTER VISION
3 credits.

Fundamentals of image analysis and computer vision; image acquisition and geometry; image enhancement; recovery of physical scene characteristics; shape-from techniques; segmentation and perceptual organization; representation and description of two-dimensional objects; shape analysis; texture analysis; goal-directed and model-based systems; parallel algorithms and special-purpose architectures. Students are strongly encouraged to have basic proficiency in calculus and linear algebra, such as MATH 340, and basic programming such as COMP SCI 300.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Develop basic computer vision applications using a programming environment
Audience: Graduate

2. Formulate computer vision research problems motivated from real-world applications
Audience: Graduate

3. Evaluate and compare existing solutions to a computer vision problem
Audience: Graduate

4. Design approaches for solving computer vision problems based on a broad range of fundamental concepts in 2D and 3D computer vision, sensing and recognition
Audience: Graduate

5. Communicate solutions verbally and in writing to justify choices while designing solutions
Audience: Graduate

COMP SCI/B M I 767 – COMPUTATIONAL METHODS FOR MEDICAL IMAGE ANALYSIS
3 credits.

Review of advanced medical image analysis techniques. Covers advanced segmentation and registration methods. Describes the use and extension of statistical and machine learning methods for medical image analysis tasks.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Implement advanced ideas from machine learning, statistics, and computer vision for medical image analysis tasks.
Audience: Graduate

2. Develop a deep technical understanding of the machine learning and statistical ideas being utilized in contemporary research in this area, and identification of blockers where research efforts can be focused
Audience: Graduate

COMP SCI 769 – ADVANCED NATURAL LANGUAGE PROCESSING
3 credits.

Develop algorithms and mathematical models for natural language processing tasks, including text categorization, information retrieval, speech recognition, machine translation, and information extraction. Focus is on the state-of-the-art computational techniques as they are applied to natural language tasks. Students are strongly encouraged to have knowledge of introductory artificial intelligence (e.g., COMP SCI 540).
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI/ED PSYCH/PSYCH 770 – HUMAN-COMPUTER INTERACTION
3 credits.

Principles of human-computer interaction (HCI); human subjects research methods and procedures, qualitative and quantitative data analysis; and semester-long research project situated in critical domains of HCI, including applications in ubiquitous, affective, assistive, social, and embodied computing.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI/B M I 771 – LEARNING BASED METHODS FOR COMPUTER VISION

3 credits.

Addresses the problems of representation and reasoning for large amounts of visual data, including images and videos, medical imaging data, and their associated tags or text descriptions. Introduces deep learning in the context of computer vision. Covers topics on visual recognition using deep models, such as image classification, object detection, human pose estimation, action recognition, 3D understanding, and medical image analysis. Emphasizes the design of vision and learning algorithms and models, as well as their practical implementations. Strongly recommended to have knowledge in computer vision or machine learning [such as COMP SCI 540] or medical image analysis [such as B M I / COMP SCI/ B M I 567].
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Demonstrate their understanding of basic theories, current approaches, key concepts, and common practices in the area of deep learning for computer vision.
Audience: Graduate

2. Recognize and distinguish among a variety of visual recognition problems in computer vision, including their problem formulations and evaluation metrics.
Audience: Graduate

3. Utilize and implement deep learning models to solve visual recognition problems.
Audience: Graduate

4. Design deep learning models for visual recognition problems, conduct experiments to evaluate the proposed model, and analyze and interpret the results.
Audience: Graduate

5. Communicate effectively through written reports, oral presentations, and discussions.
Audience: Graduate

COMP SCI 772 – LEARNING BASED IMAGE SYNTHESIS AND MANIPULATION

3 credits.

Introduces machine learning based synthesis and manipulation of visual data (images and videos). Both classical (e.g., nearest neighbor, filtering) and modern deep learning based (e.g., ConvNets, GANs, Diffusion Models) algorithms will be presented for image representation, synthesis, and manipulation. Usage of self-developed algorithms for image synthesis and manipulation to understand and analyze state-of-the-art techniques, and to identify interesting open questions and future directions.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Learning Outcomes: 1. Apply knowledge of basic theories, current approaches, key concepts, and common practices in the area of deep learning based image synthesis and manipulation.
Audience: Graduate

2. Recognize and distinguish among a variety of image and video generation problems in computer vision, including their problem formulations and evaluation metrics.
Audience: Graduate

3. Utilize and implement deep learning models to solve image and video generation problems.
Audience: Graduate

4. Design deep learning models for image and video generation problems, conduct experiments to evaluate the proposed models, and analyze and interpret the results
Audience: Graduate

5. Communicate effectively through written reports, oral presentations, and discussions.
Audience: Graduate

COMP SCI 774 – DATA EXPLORATION, CLEANING, AND INTEGRATION FOR DATA SCIENCE
3 credits.

Big Data is often said to deal with four Vs: volume, velocity, variety, and veracity. The focus is on variety and veracity challenges, which often arise in data science projects. In many such projects, data is often incorrect, hard to understand, and come from a variety of sources. Data scientists often spend 80% of their effort to explore, clean, and integrate this data, before analysis can be carried out to extract insights. As a result, managing variety and veracity has received significant attention. Study these topics, understand their challenges, and discuss solutions. These solutions often require data management, machine learning, big data scaling, cloud, crowdsourcing, and user interaction techniques. Knowledge of machine learning/AI [COMP SCI 540], databases [COMP SCI 564] and Python [COMP SCI 320] recommended.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Identify and examine the key challenges of managing variety and veracity with large data sets. These include data acquisition, data extraction, data exploration, cleaning, matching, and merging data.
Audience: Graduate

2. Summarize the variety and veracity solution approaches in academic and industry.
Audience: Graduate

3. Design and apply course concepts to experiential learning through a research project.
Audience: Graduate

4. Effectively communicate through written reports, oral presentations, and discussions.
Audience: Graduate

COMP SCI/B M I 775 – COMPUTATIONAL NETWORK BIOLOGY
3 credits.

Introduces networks as a powerful representation in many real-world domains including biology and biomedicine. Encompasses theory and applications of networks, also referred to as graphs, to study complex systems such as living organisms. Surveys the current literature on computational, graph-theoretic approaches that use network algorithms for biological modeling, analysis, interpretation, and discovery. Enables hands-on experience in network biology by implementing computational projects.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Recognize problems in molecular biology that are appropriate for network modeling
Audience: Graduate

2. Identify appropriate network algorithms for different tasks
Audience: Graduate

3. Evaluate the strengths and weaknesses of different network algorithms designed for a specific biological problem
Audience: Graduate

4. Gain practical experience in applying a select set of network algorithms on real data and evaluate its outputs
Audience: Graduate

5. Understand the algorithmic and statistical concepts of different network-based approaches
Audience: Graduate

COMP SCI/B M I 776 – ADVANCED BIOINFORMATICS
3 credits.

Advanced course covering computational problems in molecular biology. The course will study algorithms for problems such as: modeling sequence classes and features, phylogenetic tree construction, gene-expression data analysis, protein and RNA structure prediction, and whole-genome analysis and comparisons.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026
Learning Outcomes: 1. Select and implement appropriate algorithms and probabilistic models for analyzing genomes, RNA, proteins, and biological networks
Audience: Graduate

2. Write a manuscript describing a bioinformatics research study, including the motivation for the research, the scientific outcomes, and the detailed methods required to reproduce the study
Audience: Graduate

3. Read a bioinformatics research paper to identify the key computational techniques and assess the evidence supporting the paper's claims
Audience: Graduate

4. Identify emerging biological data types and data processing (e.g., single cell biology) and how the data can contribute to their research
Audience: Graduate

COMP SCI/E C E 782 – ADVANCED COMPUTER SECURITY AND PRIVACY
3 credits.

Security and privacy issues in software, networks, and hardware systems. Security vulnerabilities, privacy threats, threats modeling, and mitigation strategies. Privacy issues related to user interaction with devices, online systems, and networks. In addition, a selection of more advanced topics will be covered. Possible examples include applied cryptography in the context of systems, security and privacy policies, user authentication, and cyber-physical systems. Builds on prior experiences with one or more of the following: networking, security, modern machine learning, embedded systems, and mobile computing.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025
Learning Outcomes: 1. Identify contemporary research problems related to the security and privacy of modern computer systems
Audience: Graduate

2. Implement known security attacks to identify weaknesses that led to those attacks and evaluate defense strategies
Audience: Graduate

3. Differentiate among the different dimensions involved in protecting users' security and privacy as they relate to effectiveness, practicality, and usability
Audience: Graduate

4. Analyze, interpret, and critique research papers from top-tier security conferences by identifying their strengths and weaknesses
Audience: Graduate

5. Propose original research by defining a problem, outlining a plan, performing the original research, and designing, implementing, and evaluating the proposed solution
Audience: Graduate

6. Work effectively in teams to complete a research project
Audience: Graduate

7. Communicate effectively through written reports, oral presentations, and discussion
Audience: Graduate

COMP SCI 784 – FOUNDATIONS OF DATA MANAGEMENT
3 credits.

Foundational concepts in databases and data management. The first part of the course discusses topics on query languages (conjunctive queries, Datalog), their expressivity and complexity of evaluation. The second part studies advanced topics in modern data management, including data streams, massive parallelism, provenance, uncertain data management and privacy. There are no specific course prerequisites. It is strongly encouraged that the students are familiar with databases and relational algebra (COMP SCI 564 or equivalent). Knowledge of algorithms, complexity theory and probability will also be helpful.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026

COMP SCI 787 – ADVANCED ALGORITHMS
3 credits.

Advanced paradigms for the design and analysis of efficient algorithms, including the use of randomness, linear programming, and semi-definite programming. Applications to data structures, approximating NP-hard optimization problems, learning, on-line and distributed problems. Students are strongly encouraged to have introductory knowledge of algorithms (e.g., COMP SCI 577)
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2026

COMP SCI 790 – MASTER'S THESIS
1-9 credits.

Grad st; Master's candidates only
Requisites: Consent of instructor
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 799 – MASTER'S RESEARCH
1-9 credits.

Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 812 – ARITHMETIC ALGORITHMS
3 credits.

Survey of algorithms and design paradigms for exact arithmetic, as used in public-key cryptography, computer algebra, and pseudo-random number generation. Topics include primality testing, factorization of integers and polynomials, discrete logarithms, and (optionally) elliptic curves and integer lattices. Students are strongly encourage to have knowledge of basic abstract algebra (e.g., MATH 541), and intermediate programming ability (e.g., COMP SCI 367 or COMP SCI 300).
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2024

COMP SCI 838 – TOPICS IN COMPUTING
1-3 credits.

Advanced topics of special interest to students in various areas of Computer Science. Each offering of the course will cover a topic selected by the instructor. Credit varies by offering - check with the department to determine how an offering counts toward degree requirements.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2025

COMP SCI 839 – CORE TOPICS IN COMPUTING
3 credits.

Topics selected from advanced areas.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI/B M I/PSYCH 841 – COMPUTATIONAL COGNITIVE SCIENCE
3 credits.

Studies the biological and computational basis of intelligence, by combining methods from cognitive science, artificial intelligence, machine learning, computational biology, and cognitive neuroscience. Requires ability to program.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Spring 2025

COMP SCI/E C E/STAT 861 – THEORETICAL FOUNDATIONS OF MACHINE LEARNING

3 credits.

Advanced mathematical theory and methods of machine learning. Statistical learning theory, Vapnik-Chevronenkis Theory, model selection, high-dimensional models, nonparametric methods, probabilistic analysis, optimization, learning paradigms.
Requisites: E C E/COMP SCI 761 or E C E 830
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: No
Last Taught: Fall 2025

COMP SCI 880 – TOPICS IN THEORETICAL COMPUTER SCIENCE

3 credits.

Advanced topics in algorithms, complexity, and cryptography. The exact topic varies.
Requisites: Graduate/professional standing
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 899 – PRE-DISSERTATOR RESEARCH

1-9 credits.

Independent research supervised by a faculty member for students who have completed a master's degree but have not reached dissertator status.
Requisites: Consent of instructor
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 900 – ADVANCED SEMINAR IN COMPUTER SCIENCE

0-1 credits.

Seminar on recent research on various aspects of computer science.
Requisites: Consent of instructor
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI/B M E/B M I/BIOCHEM/CBE/GENETICS 915 – COMPUTATION AND INFORMATICS IN BIOLOGY AND MEDICINE

1 credit.

Participants and outside speakers will discuss current research in computation and informatics in biology and medicine. This seminar is required of all CIBM program trainees.
Requisites: Consent of instructor
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026
Learning Outcomes: 1. Discuss how methods from computer science, statistics, information science and engineering are applied to problems in biology, medicine and population health
Audience: Graduate

2. Recognize and be able to define applications in translational bioinformatics, clinical informatics and public health informatics
Audience: Graduate

COMP SCI 990 – DISSERTATION

1-6 credits.

Advanced level mentored reading and research for students with dissertator status.
Requisites: Consent of instructor
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2026

COMP SCI 999 – DISSERTATOR RESEARCH

1-6 credits.

Advanced level mentored reading and research for dissertators.
Requisites: Consent of instructor
Course Designation: Grad 50% - Counts toward 50% graduate coursework requirement
Repeatable for Credit: Yes, unlimited number of completions
Last Taught: Spring 2022