

CS - Computer Science

For a computer science course to be used as a prerequisite, it must have been passed with a C- or better.

Courses numbered 100 to 299 = *lower-division*; 300 to 499 = *upper-division*; 500 to 799 = *undergraduate/graduate*.

CS 211. Introduction to Programming (4).

First course in computer programming in a high-level language. Emphasizes analyzing problems, designing solutions and expressing them in the form of a well-structured program using the procedural aspects of C++. Prerequisite(s): MATH 111. Corequisite(s): CS 211L.

CS 281I. Noncredit Internship (0).

Complements and enhances the student's academic program by providing an opportunity to apply and acquire knowledge in a workplace environment as an intern. Prerequisite(s): departmental consent.

CS 311. Object-Oriented Programming (4).

Concepts of object-oriented programming. Covers data abstractions, classes and objects, methods, inheritance, polymorphism, dynamically-bound method calls and data encapsulation. Includes programming assignments in C++. Prerequisite(s): CS 211. Corequisite(s): CS 311L.

CS 321. Discrete Structures I (3).

Cross-listed as MATH 321. Provides a mathematical foundation essential to the entire computer science curriculum. Includes propositional and predicate logic, induction, recursion and counting techniques. Prerequisite(s): MATH 242 or equivalent with a grade of 2.000 or better.

CS 322. Discrete Structures II (3).

Cross-listed as MATH 322. Continuation of Discrete Structures I. Includes relations, graphs, trees, Boolean algebra and automata. Prerequisite(s): MATH 321 or CS 321.

CS 331. Introduction to Powershell for Hackers (1).

Cross-listed as AC 331. Students learn the basics of using Windows PowerShell and how to interact with Windows via the PowerShell command prompt. Course instruction includes: how to create and interact with Windows files and directories, how to interact with Windows processes and services, how to use PowerShell commands for basic digital forensics analysis, and how to create and use basic PowerShell scripts. Repeatable for credit.

CS 352. Competitive Ethical Hacking (3).

Cross-listed as AC 352. Presents fundamental concepts of competitive ethical hacking in computer and network security. The course introduces the command line interface, open-source intelligence, cryptography, digital forensics, web application security, software reverse engineering, secure programming and log analysis. Assignments include participating in capture the flag competitions. Prerequisite(s): CS 211 or (AC 121 and AC 322).

CS 356. Introduction to Computer Security (3).

Provides a first approach to the security mindset. Covers how adversaries exploit common software vulnerabilities such as buffer overflows and race conditions. Students learn secure coding techniques and countermeasures. Topics also include cryptography principles, web security, and legal and ethical aspects. Prerequisite(s): CS 211.

CS 400. Data Structures (4).

Introduces basic data structures and covers their implementations using classes in C++. Includes lists, stacks, queues, binary trees and hash tables. Prerequisite(s): CS 311 with a C- grade or better. Corequisite(s): CS 400L.

CS 400L. Data Structures Lab I (0).

Applies concepts from CS 400 in a lab setting. Corequisite(s): CS 400.

CS 410. Programming Paradigms (3).

Overview of different programming paradigms, including their philosophies, uses and relative advantages/disadvantages. Covers the procedural/imperative, functional, logic and object-oriented paradigms. Includes programming assignments in the functional and logic paradigms. Prerequisite(s): CS 311.

CS 481. Cooperative Education (1-3).

Provides a field placement that integrates theory with a planned and supervised professional experience designed to complement and enhance the student's academic program. Individualized programs must be formulated in consultation with, and approved by, appropriate faculty sponsors. Repeatable for credit. Prerequisite(s): departmental consent.

CS 481I. Noncredit Internship (0).

Complements and enhances the student's academic program by providing an opportunity to apply and acquire knowledge in a workplace environment as an intern. Prerequisite(s): departmental consent.

CS 481N. Internship (1-3).

Complements and enhances the student's academic program by providing an opportunity to apply and acquire knowledge in a workplace environment as an intern. Prerequisite(s): departmental consent.

CS 497. Special Topics (1-3).

An umbrella course created to explore a variety of subtopics differentiated by letter (e.g., 497A, 497B). Not all subtopics are offered each semester – see the course schedule for availability. Students enroll in the lettered courses with specific topics in the titles rather than in this root course. Prerequisite(s): departmental consent.

CS 497Q. Introduction to the Linux Operating System (1).

Students learn basic Linux tools, Linux administration, Linux file and directory structure, and related concepts.

CS 498. Individual Projects (1-3).

Arranged individual independent project in specialized content areas under the supervision of a faculty member. Repeatable for a total of 6 credit hours. Prerequisite(s): departmental consent.

CS 510. Programming Language Concepts (3).

Theoretical concepts in the design and use of programming languages. Formal syntax, including Backus Normal Form (BNF), Extended Backus-Naur Form (EBNF), and syntax diagrams. Semantics, including declaration, allocation and evaluation, symbol table and runtime environment; data types and type checking, procedure activation and parameter passing, modules and abstract data types. Prerequisite(s): CS 311, MATH 322.

CS 540. Operating Systems (3).

Fundamental principles of modern operating systems. CPU management including processes, threads, scheduling, synchronization, resource allocation and deadlocks. Memory management including paging and virtual memory. Storage management and file systems. Prerequisite(s): ECE 238 and CS 311.

CS 560. Design and Analysis of Algorithms (3).

Design of various algorithms including several sorting algorithms. Analysis of their space and time complexities. Data structures include heaps, hash tables and binary search trees. Prerequisite(s): CS 322, 400; STAT 460 or IME 254.

CS 577. Special Topics in Computer Science (1-4).

Focuses on contemporary computer science topics through traditional lecture, research and experiential learning activities. Content changes as new problems and research advances related to computer science attain prominence nationally and internationally. Prerequisite(s): departmental consent.

CS 580. Introduction to Software Engineering (3).

Introduces the processes, methods and tools used in software development and maintenance. Topics include software development life cycle and processes, configuration management, requirements gathering, OOA/D with UML, cohesion and coupling, and unit testing. Prerequisite(s): CS 311.

CS 581. Programming for Computing (3).

Fast-paced course in computer programming in Python. Emphasizes problem solving and object-oriented programming principles. Topics include language syntax, built-in data structures, functions, classes, modules and exception handling. Not for BSCE, BSCS or BSEE credit. Prerequisite(s): MATH 242.

CS 582. Applied Data Structure and Algorithms for Computing (3).

Fast-paced course that introduces basic data structures, algorithms, and introduces analysis of space and time complexity. Topics includes sorting algorithms, greedy algorithms, linked lists, stacks, queues, binary search trees, asymptotic notation, substitution and recursion tree method, master theorem, and order statistics. Not for BSCE, BSCS or BSEE credit. Prerequisite(s): CS 581.

CS 598. Senior Design Project I (2).

Cross-listed as ECE 585. Design project under faculty supervision chosen according to the student's interest. Does not count toward a graduate degree in electrical engineering, computer engineering or computer science. This class should be taken in the semester prior to the one in which the student is going to graduate. For undergraduate credit only. Prerequisite(s): senior standing, ECE 492 or CS 580. Pre- or corequisite(s): PHIL 354 or PHIL 385.

CS 599. Senior Design Project II (2).

This is the second part of a sequence of two courses (CS 598 and CS 599) that have to be taken in two consecutive semesters. Students failing this course must retake the CS 598 course. For undergraduate credit only. Prerequisite(s): CS 598. Pre- or corequisite(s): PHYS 314 and PHYS 316.

CS 656. Introduction to Cybersecurity (3).

Provides fundamental concepts in cybersecurity including cryptography, security in networks, operating systems and databases. Topics in intrusion detection, security administration, and legal and ethical issues are also discussed. Prerequisite(s): CS 664 or CS 764.

CS 664. Computer Networks (3).

Introductory course on computer networking. Introduces concepts and protocols in various network layers with emphasis on the internet. Topics covered include: physical layer (wired and wireless), medium access control and data link layers, packet switching and routing (IP), routing protocols, transport layer (TCP, UDP), congestion and flow control, basic network security, and network applications. Prerequisite(s): undergraduate students: IME 254 and CS 311; graduate students: object oriented programming and statistics/probability knowledge.

CS 665. Introduction to Database Systems (3).

Fundamental aspects of relational database systems, conceptual database design and entity-relationship modeling; the relational data model and its foundations, relational languages and SQL, functional dependencies and logical database design; views, constraints and

triggers. Course includes a group project involving the design and implementation of a relational database and embedded SQL programming. Prerequisite(s): CS 311, MATH 322.

CS 697. Selected Topics (1-3).

An umbrella course created to explore a variety of subtopics differentiated by letter (e.g., 697A, 697B). Not all subtopics are offered each semester – see the course schedule for availability. Students enroll in the lettered courses with specific topics in the titles rather than in this root course. Prerequisite(s): departmental consent.

CS 697AP. Applied Parallel Computing (3).

This course is to teach how to program parallel computers to efficiently analyze challenging problems with enormous datasets. Two distinct approaches will be introduced which can be used to solve problems in all manner of domains including data analytics and machine learning. The first approach to be studied will be embarrassingly parallel in nature while the second approach will leverage fine-grain parallelism. Prerequisite(s): CS 394 or Instructor's consent.

CS 697AQ. Web Programming (3).

Hands-on introduction to web programming. Prepares students to create webpages and develop web applications that integrate with a backend database. Topics covered include client-side technologies that run in the web browser (HTML, CSS and JavaScript), and server-side technologies that run on the web server (Node.js or PHP and SQL). A strong programming background is preferred for successful completion of several practical exercises contained in the course. Prerequisite(s): CS 311.

CS 697AR. Power Business Intelligence (3).

Covers applications of data science with a focus on real-life examples from various industries. Technical skills, soft skills and subject matter expertise are developed during this course as the intersection of these skills is critical to the field of data science. Students are introduced to different software and programming languages. Topics include project scoping; collection, analysis and visualization of data; interpreting and communicating results; teamwork; and problem-solving. Additionally, students have an opportunity to use their interests and subject matter expertise to interpret findings, communicate results and offer solutions.

CS 721. Advanced Algorithms and Analysis (3).

Topics include height-balanced trees, graph algorithms, greedy algorithms, dynamic programming, hard problems and approximation algorithms. Prerequisite(s): CS 560.

CS 731. Mathematical Foundations for Computer Networking (3).

Introductory class on applying various mathematical tools to the field of computer networks and related areas. Divided into three phases: phase one covers the fundamentals of probability, statistics and linear algebra required for understanding the core topics to follow. Phase two covers the core topics of optimization and queuing theory. Phase three briefly covers the advanced topics of game theory and information theory. The depth of coverage is sufficient to allow students to read and understand research papers in computer networking and related areas that use these standard techniques. Ideas are taught through intuition, mathematically correct formalization and detailed numerical examples. Prerequisite(s): MATH 243. Corequisite(s): CS 664.

CS 737. Wireless Networking (3).

Covers topics ranging from physical layer to application layer in the wireless and mobile networking fields. Explores physical layer issues of wireless communications, wireless cellular telephony, ad-hoc networks, mobile IP and multicast, wireless LAN (IEEE 802.11), security, Bluetooth and WAP, etc. Imparts general knowledge about wireless communication technologies and ongoing research activities. Prerequisite(s): CS 664.

CS 746. Perspectives on Data Science (3).

Covers the perspectives and fundamentals of data science. Topics include data collection, preprocessing, transformation, exploratory data analysis, visualization, predictive modeling, descriptive modeling, clustering, regression and classification and data science project life cycle. This course is limited to engineering students and students in other colleges majoring in data science/analytics related programs. Prerequisite(s): IME 254 and CS 211 for undergraduate students; instructor's consent for graduate students.

CS 764. Routing and Switching I (3).

Introductory course studying the operation of layer-2 architectures such as vlans, spanning tree protocol, and the operation of layer-3 routing protocols such as OSPF, ISIS, and BGP. Prerequisite(s): CS 664.

CS 767. Foundations of Network Security (3).

Presents fundamental concepts in cryptography and network security, and discusses applications and protocols for providing confidentiality, authentication, integrity and availability in networking services and systems. Includes review of symmetric-key cryptographic schemes such as DES and AES, public-key cryptographic schemes such as RSA and Diffie-Hellman key exchange protocol, cryptographic hash functions such as SHA, message authentication codes such as HMAX, digital signature schemes such as El-Gamal and DSS, kerberos and user authentication protocols, transport layer security and TLS, IP layer security and IPSec, and wireless security principles. Prerequisite(s): CS 664; the department also highly recommends CS 656, but it is not required.

CS 770. Machine Learning (3).

Introductory course on machine learning. Topics include supervised and unsupervised learning, k-nearest neighbor classifiers, decision trees, linear regression, support vector machines, kernel methods, naïve Bayes classifier, logistic regression, neural networks, deep learning, k-means clustering, hierarchical clustering, mixture of Gaussians and EM algorithms, linear and non-linear dimension reductions. Prerequisite(s): MATH 511, IME 254 and CS 311; or CS 746.

CS 771. Artificial Intelligence (3).

Introduces some of the fundamental concepts and techniques underlying artificial intelligence. Topics covered include state spaces, heuristic search, game playing, knowledge representation, and resolution in propositional and first-order predicate logic. Prerequisite(s): CS 560.

CS 780. Advanced Software Engineering (3).

Systematically learns and practices all workflows of the Unified Process. Discusses advanced topics in software development, maintenance and evolution. Topics include software design patterns, architecture and architectural styles, frameworks, refactorings, and static and dynamic analyses. Includes a group project. Prerequisite(s): CS 580.

CS 781. Cooperative Education (1-3).

Practical experience in a professional environment to complement and enhance the student's academic program. For master's level CS students. Repeatable for credit, but may not be used to satisfy degree requirements. Prerequisite(s): departmental consent and graduate GPA of 3.000 or above.

CS 796. Network Programming (4).

Introduces techniques for developing TCP and UDP network clients, servers and applications. Topics covered include sockets, client/server design alternatives, concurrent processes and threads, web applications, and security. Programming-intensive course that assumes some experience with programming in a high-level language.

CS 797. Special Topics in Computer Science (1-4).

An umbrella course created to explore a variety of subtopics differentiated by letter (e.g., 797A, 797B). Not all subtopics are offered each semester – see the course schedule for availability. Students enroll in the lettered courses with specific topics in the titles rather than in this root course. Prerequisite(s): departmental consent.

CS 797J. Machine Learning for Natural Language Processing (3).

Provides students the opportunity to learn about machine learning for Natural Language Processing (NLP). NLP is the field of study that is related to computational processing/manipulation of speech and text. This course deals with introduction to various computational tools/algorithms for text processing, predictive text, automatic summarization, sentiment analysis, information extraction, POS tagging and other related NLP tasks. In addition to providing introduction to various algorithms for analyzing the content and structure of texts, the course deals with algorithms/tools to extract information from unstructured text. As present day NLP is highly intertwined with machine learning, this course also provides introduction to various machine learning algorithms utilized in the field of NLP. A brief introduction of deep learning approaches for NLP are also presented. The course also introduces techniques to prepare text data for modeling. Students get hands-on experience in working with a capstone final project. Prerequisite(s): CS 746 or instructor's consent.

CS 797K. Advanced Topics in Data Storage (3).

Examines the history of computer data storage, the current state of the technology, and look at possible future developments. Examines RAID storage concepts, file systems, databases, object storage and cloud storage. The course is experimental and exploratory; student interest drives topics and research projects.

CS 797M. Introduction to Linear Data Modeling (3).

Introduction to commonly used linear models and modeling techniques to be able to reason with data. Topics include linear regression techniques and using such techniques with higher order data. Implementation and application of the models are performed via examples using Python and R. Prerequisite(s): CS 400.

CS 797N. Data Visualization (3).

Data visualization is the art and science of turning data into readable graphics. Students explore how to design and create data visualizations based on data available and tasks to be achieved. This process includes data modeling, data processing, mapping data attributes to graphical attributes, and strategic visual encoding based on known properties of visual perception as well as the task(s) at hand. Students also learn to evaluate the effectiveness of visualization designs and think critically about each design decision, such as choice of color and choice of visual encoding. Students create their own data visualizations and learn to use data visualization tools. Prerequisite(s): CS 400.

CS 797O. Neural Networks and Deep Learning (3).

Provides students with in depth knowledge of the use, structure, implementation and analysis of neural networks. The main topics covered are structure and basic types of neural networks, learning techniques, and specialized networks, including but not limited to Hopfield networks, dynamic and competitive networks, Grossberg Networks, and recurrent networks. Case studies are also used to understand the use and optimization with real world examples.

CS 797P. Algorithms and Applications on Graphs (3).

Provides students with in depth knowledge of the use, structure, implementation and applications of using graphs to represent data. The main topics covered are basic graph theory, breadth first and depth first algorithms, spanning tree algorithms, minimum cut/max flow, modulus, and more.

CS 797Q. Applied and Practical Data Science (3).

Provides a practical view of data science and its industry applications. Topics discussed include: data science methodology, data science project management, interpreting data, data informed recommendation making, data visualization and model iteration. Prerequisite(s): CS 746.

CS 797R. Cross-Platform Mobile Development (3).

A project-based course that teaches fundamental factors of cross-platform (Android, iOS, Windows desktop) mobile application development. It uses .NET Multi-platform App UI (.NET MAUI) as the framework to implement applications. Students learn user interface elements, data binding, data persistence and model-view-view-model (MVVM) design pattern. Prerequisite(s): graduate standing or CS 400.

CS 797S. Software Quality Engineering (3).

Designed to provide computer science students broad view of software quality assurance techniques and their integration into a comprehensive quality management and improvement system. Course coverage includes topics outlined in key ISO and IEEE Standards. These include the benefits of software quality, defect prevention and detection techniques; an overview of software quality-related standards and models; and a review of major quality assurance activities in the software development life cycle. The course explores contemporary topics on teams and project management principles. Prerequisite(s): CS 580.

CS 797T. Data Cleaning (3).

Before someone can start testing machine learning algorithms, they need clean data. It is estimated that data scientists spend up to 80 percent of their time cleaning data before they get to the analysis. Spending time exploring, understanding and cleaning data is paramount. This course gives students practical experience with various data-cleaning procedures and exploratory data analysis techniques. Students learn the best practices and dive further into the basics with hands-on examples. This course teaches students how to identify and treat a variety of data-cleaning problems in Python.

CS 797U. Advanced Information Systems (3).

Designed to provide computer science students understanding of information systems in a modern enterprise, including database design, information technology and ethics using hands-on activities and directed classroom discussion. The class concentrates on the relational database model; the conceptual, logical and physical phases of database design and development; and entity-relationship modeling, data normalization. Students gain hands-on experience with database design concepts. The opportunity to build a relational database using the foundational principles of database design within the framework of the database life cycle (DBLC) is covered. Additionally, the course emphasizes the relational data model and its foundations, relational languages and SQL, functional dependencies, and logical database design. Prerequisite(s): CS 211 or MIS 310 or MATH 451 or CS 497B.

CS 797V. Artificial Intelligence for Cybersecurity (3).

Covers advanced topics in the application of AI to defend and attack systems, including AI systems themselves. The course starts by covering the implementation of machine learning algorithms to defend computer and networks against common attacks. It then covers the AI techniques that attackers can use to craft attacks to bypass AI defenses, as well as potential countermeasures. A major part of the course is a project that implements both AI defenses and attacks for a particular system. Prerequisite(s): CS 770 and (CS 656 or CS 767).

CS 797W. Quantum Computing Essentials and Applications (3).

Quantum computing is the future. In this course, students dive into the mysterious and complex quantum world. By understanding the key difference between classical computing and quantum computing and learning the essentials of quantum computing, students explore

potential ideas to incorporate elements of quantum computing into their research areas. This project-based course provides students with an opportunity to experiment and create original codes working with real quantum computers at IBM Quantum Computing and Google Quantum AI. This hands-on experience further develops an understanding of the applications of quantum computers. The course also improves the much-needed research and presentation skills of a graduate student. The course focuses on a brief history of quantum computing, the concept of superposition, qubits and their behavior, the concept of entanglement, quantum teleportation, and EPR paradox. Prerequisite(s): MATH 511 and CS 211.

CS 798. Individual Projects (1-3).

Allows beginning graduate students and mature undergraduate students to pursue individual projects of current interest in computer science. Repeatable for credit with advisor approval. Prerequisite(s): departmental consent.