Computer Science

**CS 1134** Computer Science Laboratory (1 semester credit hour) Laboratory course to accompany **CS 1334**. This course assists students in experiencing elementary programming in a high-level language. May not be used to satisfy degree requirements for majors in the School of Engineering and Computer Science. Credit cannot be received for both courses, **CS 1134** and **CS 1136**. Corequisite: **CS 1334**. (0-3) S

**CS 1136** Computer Science Laboratory (1 semester credit hour) Laboratory course to accompany **CS 1336**. This course assists students in experiencing elementary programming in a high-level language. May not be used to satisfy degree requirements for majors in the School of Engineering and Computer Science. Corequisite: **CS 1336**. (0-3) S

**CS 1200** Introduction to Computer Science and Software Engineering (2 semester credit hours) Introduction to the computing professions; overview of Computer Science (CS) and Software Engineering (SE) curricula, connections with Computer Engineering, other Engineering and Computer Science fields, and Arts and Technology programs; problem solving and other skills needed to succeed as a CS or SE major. Introduction to quantitative methods; team projects designed to replicate decision processes and problem solving in real-world situations; additional preparatory topics for CS and SE majors. **BMEN 1100** or **CE 1100** or **EE 1100** or **MECH 1100** can substitute for this course (together with 1 hour of CS elective). Credit cannot be received for more than one of the following: **BMEN 1100** or **CE 1100** or **CS 1200** or **EE 1100** or **MECH 1100**. (2-0) Y

**CS 1324** Introduction to Programming for Biomedical Engineers (3 semester credit hours) Computer programming in a high-level, block structured language with a focus on engineering applications in medicine. Basic data types and variables, memory usage, control structures, functions/procedures and parameter passing, recursion, input/output. Programming projects related to biomedical engineering applications. May not be used to satisfy degree requirements for majors in Computer Engineering, Computer Science, Software Engineering, and Telecommunications Engineering. Prerequisite: **CS 1336** or equivalent. (3-0) S

**CS 1325** Introduction to Programming (3 semester credit hours) Computer programming in a high-level, block structured language. Basic data types and variables, memory usage, control structures, functions/procedures and parameter passing, recursion, input/output. Programming projects related to engineering applications, numerical methods. May not be used to satisfy degree requirements for majors in Computer Engineering, Computer Science, Software Engineering, and Telecommunications Engineering. Prerequisite: **CS 1336** or equivalent. (3-0) S

**CS 1334** Programming Fundamentals for Non-Majors (3 semester credit hours) Introduction to computers. Primitive data types, variable declarations, variable scope, and primitive operations. Control statements. Methods/functions. Arrays and strings using primitive data arrays. Output formatting. Debugging techniques. Designed for students with no prior computer programming experience. May not be used to satisfy degree requirements for majors in the School of Engineering and Computer Science. Credit cannot be received for both courses, **CS 1334** and **CS 1336**. Note that a grade of C or better is required in order to register for **CS 1335**. Corequisite: **CS 1134**. (3-0) S

**CS 1335** Computer Science I for Non-majors (3 semester credit hours) Introduction to object-oriented software analysis, design, and development. Classes and objects. Object composition and polymorphism.
Sorting and searching. Strings using core classes. Inheritance and interfaces. Graphical User Interfaces. May not be used to satisfy degree requirements for majors in the School of Engineering and Computer Science, especially majors in Computer Science and Engineering. Credit cannot be received for both courses, \textit{CS 1335} and \textit{(CE 1337 or CS 1337 or TE 1337)}. Prerequisite: \textit{CS 1334} with a grade of C or better or equivalent. (3-0) S

\textbf{CS 1336 (COSC 1336)} Programming Fundamentals (3 semester credit hours) Introduces the fundamental concepts of structured programming. Topics include software development methodology, data types, control structures, functions, arrays, and the mechanics of running, testing, and debugging. Programming language of choice is C. The class is open to students in the School of Engineering and Computer Science only. May not be used to satisfy degree requirements for majors in the School of Engineering and Computer Science. Note that a grade of C- or better in this class is required in order to register for \textit{(CS 1324 or CS 1325)}; a grade of C or better in this class is required to register for \textit{(CE 1337 or CS 1337 or TE 1337)}. Corequisite: \textit{CS 1136}. (3-0) S

\textbf{CS 1337 (COSC 1337)} Computer Science I (3 semester credit hours) Review of control structures and data types with emphasis on structured data types. Applies the object-oriented programming paradigm, focusing on the definition and use of classes along with the fundamentals of object-oriented design. Includes basic analysis of algorithms, searching and sorting techniques, and an introduction to software engineering. Programming language of choice is C/C++. Students will also be registered for an exam section. Prerequisite: \textit{CS 1336} with a grade of C or better or equivalent. (Same as \textit{CE 1337} and \textit{TE 1337}) (3-0) S

\textbf{CS 2305 (MATH 2305)} Discrete Mathematics for Computing I (3 semester credit hours) Principles of counting. Boolean operations. Logic and proof methods. Recurrence relations. Sets, relations, functions. Elementary graph theory. Elementary number theory. Prerequisite: ALEKS score required or \textit{MATH 2312} with a grade of C or better. (Same as \textit{CE 2305} and \textit{TE 2305}) (3-0) S

\textbf{CS 2335} Computer Science II for Non-majors (3 semester credit hours) Exceptions and number formatting. File input/output using Stream classes. Implementation of primitive data structures, including linked lists, stacks, queues, and binary trees. Advanced data manipulation using core classes. May not be used to satisfy degree requirements for majors in the School of Engineering and Computer Science. Credit cannot be received for both courses, \textit{CS 2335} and \textit{(CE 2336 or CS 2336 or TE 2336)}. Prerequisite: \textit{CS 1335} or \textit{CE 1337} or \textit{CS 1337} or \textit{TE 1337}. (3-0) S

\textbf{CS 2336 (COSC 2336)} Computer Science II (3 semester credit hours) Further applications of programming techniques, introducing the fundamental concepts of data structures and algorithms. Topics include recursion, fundamental data structures (including stacks, queues, linked lists, hash tables, trees, and graphs), and algorithmic analysis. Includes comprehensive programming projects. Programming language of choice is Java. Credit cannot be received for both \textit{CS 2337} and \textit{(CS 2336 or CE 2336 or TE 2336)}. Prerequisite: \textit{CE 1337} or \textit{CS 1337} or \textit{TE 1337} with a grade of C or better. Prerequisite or Corequisite: \textit{CE 2305} or \textit{CS 2305} or \textit{TE 2305} with a grade of C or better. (Same as \textit{CE 2336} and \textit{TE 2336}) (3-0) S

\textbf{CS 2337} Computer Science II (3 semester credit hours) Further applications of programming techniques, introducing the fundamental concepts of data structures and algorithms. Topics include recursion, fundamental data structures (including stacks, queues, linked lists, hash tables, trees, and graphs), and algorithmic analysis. Includes comprehensive programming projects. Programming language of choice is C++. Credit cannot be received for both \textit{CS 2337} and \textit{(CS 2336 or CE 2336 or TE 2336)}. Prerequisite: AP score of at least 4. Prerequisite or Corequisite: \textit{CE 2305} or \textit{CS 2305} or \textit{TE 2305} with a grade of C or better. (3-0) S

\textbf{CS 2V95} Individual Instruction in Computer Science (1-6 semester credit hours) Individual study under a faculty member's direction. May be repeated for credit as topics vary (6 semester credit hours maximum).
Instructor consent required. ([1-6]-0) R

**CS 3149** Competitive Learning in Computer Science (1 semester credit hour) In this course, students will work together in small teams to solve graduated problems, similar to those used in programming contests around the world. Approaches to categorizing problems and selecting appropriate data structures and algorithms will be covered, along with types of algorithms for solving problems (brute force, greedy, divide and conquer, dynamic programming). Students will do problem solving in a competitive environment against the clock. May be repeated for credit as topics vary (3 semester credit hours maximum). Prerequisites: (CE 2336 or CS 2336 or TE 2336) with a grade of C or better and CS 3305 with a grade of C or better. (1-0) Y

**CS 3162** Professional Responsibility in Computer Science and Software Engineering (1 semester credit hour) Professional and ethical responsibilities of computer scientists and software engineers as influenced by growth in computer use and networks. Costs and benefits of computer technology. Risks and liabilities of safety-critical systems. Social implications of the Internet. Interaction between human values and technical decisions involving computing. Intellectual Property. Global impact of computing. Prerequisites or Corequisites: CS 3345 and CS 3354 and ECS 2361. (Same as SE 3162) (1-0) S

**CS 3305** Discrete Mathematics for Computing II (3 semester credit hours) Advanced counting methods; recurrence relations, divide and conquer algorithms, principle of inclusion and exclusion. Partial orders and lattices, Algorithmic complexity. Graph theory. Strings and languages. Number theory. Elements of modern algebra. Credit cannot be received for both courses, CS 3305 and SE 3306. Double majors are required to take CS 3305. Prerequisites: (CE 2305 or CS 2305 or TE 2305) with a grade of C or better, and (MATH 2414 or MATH 2419). (3-0) S

**CS 3333** Data Structures (3 semester credit hours) Programming with basic data structures (arrays, stacks, queues, lists, and trees) and their associated algorithms. Various sorting and searching techniques. Fundamental graph algorithms. This course covers much of the same material as CS 3345 without requiring the analysis of algorithms. May not be used to satisfy degree requirements for majors in Computer Science. Credit cannot be received for both courses, (CE 2336 or CS 2336 or TE 2336) and CS 3333. Prerequisite: CS 1335 or (CE 1337 or CS 1337 or TE 1337) or CS 3335 or equivalent programming experience. (3-0) Y

**CS 3335** C and C++ (3 semester credit hours) Numerous programming projects in both C and C++. All fundamentals of C, with special emphasis on use of pointers. Use of C++ extensions to create and extend (by inheritance) abstract data types. The use/advantages of virtual functions (dynamic polymorphism). Prerequisite: CS 2335 or (CE 2336 or CS 2336 or TE 2336) or equivalent. (3-0) T

**CS 3340** Computer Architecture (3 semester credit hours) This course introduces the concepts of computer architecture by going through multiple levels of abstraction, and the numbering systems and their basic computations. It focuses on the instruction-set architecture of the MIPS machine, including MIPS assembly programming, translation between MIPS and C, and between MIPS and machine code. General topics include performance calculation, processor datapath, pipelining, and memory hierarchy. Credit cannot be received for both courses, (CS 3340 or SE 3340 or TE 3340) and (CE 4304 or EE 4304). Prerequisites: (CE 1337 or CS 1337 or TE 1337 with a grade of C or better or equivalent) and (CE 2305 or CS 2305 or TE 2305 with a grade of C or better). (Same as SE 3340 and TE 3340) (3-0) S

**CS 3341** Probability and Statistics in Computer Science and Software Engineering (3 semester credit hours) Axiomatic probability theory, independence, conditional probability. Discrete and continuous random variables, special distributions of importance to CS/SE, and expectation. Simulation of random variables and Monte Carlo methods. Central limit theorem. Basic statistical inference, parameter estimation,
hypothesis testing, and linear regression. Introduction to stochastic processes. Illustrative examples and simulation exercises from queuing, reliability, and other CS/SE applications. Credit cannot be received for both courses, (CS 3341 or SE 3341 or STAT 3341) and ENGR 3341. Prerequisites: (MATH 1326 or MATH 2414 or MATH 2419), and (CE 2305 or CS 2305 or TE 2305 with a grade of C or better). (Same as SE 3341 and STAT 3341) (3-0) S

CS 3345 Data Structures and Introduction to Algorithmic Analysis (3 semester credit hours) Analysis of algorithms including time complexity and Big-O notation. Analysis of stacks, queues, and trees, including B-trees. Heaps, hashing, and advanced sorting techniques. Disjoint sets and graphs. Course emphasizes design and implementation. Prerequisites: (CE 2305 or CS 2305 or TE 2305 with a grade of C or better) and (CE 2336 or CS 2336 or TE 2336 with a grade of C or better). Prerequisite or Corequisite: (CS 3341 or SE 3341 or ENGR 3341). (Same as CE 3345 and SE 3345 and TE 3345) (3-0) S

CS 3354 Software Engineering (3 semester credit hours) Introduction to software life cycle models. Software requirements engineering, formal specification and validation. Techniques for software design and testing. Cost estimation models. Issues in software quality assurance and software maintenance. Prerequisites: (CE 2336 or CS 2336 or TE 2336 with a grade of C or better or CS 3333) and (CE 2305 or CS 2305 or TE 2305 with a grade of C or better or equivalent). Prerequisite or Corequisite: ECS 3390. (Same as CE 3354 and SE 3354) (3-0) Y

CS 3360 Computer Graphics for Artists and Designers (3 semester credit hours) Device and logical coordinate systems, and the nature of raster display. Algorithms for basic 2-D drawing primitives, such as line-drawing, clipping and Bezier curves. Perspectives in 3-D, and hidden-face elimination, such as Painter’s and Z-Buffer algorithms. Color and texture. Fractals and the Mandelbrot set. May not be used to satisfy degree requirements for majors in the School of Engineering and Computer Science. Prerequisite: CS 2335. (3-0) Y

CS 3377 C/C++ Programming in a UNIX Environment (3 semester credit hours) Advanced programming techniques utilizing procedural and object oriented programming in a UNIX environment. Topics include basic UNIX concepts, file input and output, implementation of strings, and dynamic memory allocation/management. Design and implementation of a comprehensive programming project is required. Prerequisite: (CE 2336 or CS 2336 or TE 2336) with a grade of C or better or equivalent. (Same as SE 3377) (3-0) S

CS 3385 Ethics, Law, Society, and Computing (3 semester credit hours) Issues of professional ethics; computer crime; wiretapping and encryption; protecting software and other intellectual property; privacy and information; careers and computers; reliability and safety; constitutional issues. Broader issues on the impact and control of computers. (3-0) S

CS 3V95 Undergraduate Topics in Computer Science (1-9 semester credit hours) Subject matter will vary from semester to semester. May be repeated for credit as topics vary (9 semester credit hours maximum). ([1-9]-0) S

CS 4141 Digital Systems Laboratory (1 semester credit hour) Laboratory to accompany CS 4341. The purpose of this laboratory is to give students an intuitive understanding of digital circuits and systems. Laboratory exercises include construction of simple digital log circuits using prototyping kits and board-level assembly of a personal computer. Corequisite: CS 4341 or TE 4341. (Same as TE 4141) (0-3) S

CS 4301 Special Topics in Computer Science (3 semester credit hours) May be repeated for credit as topics vary (9 semester credit hours maximum). Prerequisite: CS 3345 or CS 3345 or SE 3345 or TE 3345. (3-0) S

https://catalog.utdallas.edu/2019/undergraduate/courses/cs
CS 4313 Neural Net Mathematics (3 semester credit hours) Advanced matrix calculus and vector calculus-based probability theory with applications to problems in machine learning and artificial neural network modeling. Intended to provide mathematics preparation for CGS 4314 or CS 4314 and CGS 4315 or CS 4315. Includes introduction to gradient descent type unsupervised, supervised, and reinforcement learning algorithms as well as iterative constraint satisfaction algorithms. Prerequisites: (MATH 2414 or MATH 2419) and (STAT 3341 or STAT 4351) and MATH 2418. (Same as CGS 4313) (3-0) T

CS 4314 Intelligent Systems Analysis (3 semester credit hours) This course covers mathematics essential for the mathematical analysis and design of unsupervised, supervised, and reinforcement machine learning algorithms including Neural Network learning machines within a statistical empirical risk minimization framework. Course topics include: advanced vector and matrix calculus, stochastic sequences of mixed random vectors, and the Markov random field factorization theorem with explicit machine learning applications and examples. Prerequisite: CGS 4313 or instructor consent required. (Same as CGS 4314) (3-0) T

CS 4315 Intelligent Systems Design (3 semester credit hours) Mathematical analysis of behavior and generalization performance of deterministic batch and stochastic adaptive learning algorithms within a statistical empirical risk minimization framework. Topics include: Convergence analysis of batch learning algorithms, convergence analysis of adaptive learning algorithms, Comte Carlo Markov Chain inference and sampling, bootstrap sampling methods, and estimation of generalization performance using asymptotic statistical theory. Unsupervised, supervised, and reinforcement machine learning applications are emphasized throughout the course. Prerequisite: CGS 4313 or instructor consent required. (Same as CGS 4315) (3-0) T

CS 4332 Introduction to Programming Video Games (3 semester credit hours) Video game programming concepts. Programming with game engine. 2D and 3D computer graphics techniques and data structures. Computer animation, physics-based methods and collision detection. GPU and shader programming. Artificial intelligence for video games. Networking and multiplayer. Prerequisite: CE 3345 or CS 3345 or SE 3345 or TE 3345. (3-0) Y

CS 4334 Numerical Analysis (3 semester credit hours) Solution of linear equations, roots of polynomial equations, interpolation and approximation, numerical differentiation and integration, solution of ordinary differential equations, computer arithmetic, and error analysis. Prerequisites: (MATH 2370 or CS 1324 or CS 1325 or CE 1337 or CS 1337 or TE 1337) and (MATH 2418 and MATH 2451). (Same as MATH 4334) (3-0) Y

CS 4336 Advanced Java (3 semester credit hours) Advanced Java programming techniques for enterprise application development. Covers Java Enterprise API's for working with databases, web servers, and application servers. Students will create multi-tiered web applications and web services integrated with a database. Prerequisite: CE 2336 or CS 2336 or TE 2336 or equivalent. (3-0) T

CS 4337 Organization of Programming Languages (3 semester credit hours) Principles of design and implementation of contemporary programming languages. Formal description including specification of syntax and semantics of programming languages. Language definition structures including binding, scoping, data types, control structures, parameter passing, abstraction mechanism, and run-time considerations. Design issues of imperative languages, object-oriented languages, functional languages and logic languages. Design, implement, and debug programs in various programming language paradigms. Prerequisites: (CE 2336 or CS 2336 or TE 2336) with a grade of C or better or CS 3333 and (CE 2305 or CS 2305 or TE 2305) with a grade of C or better and (CS 3340 or SE 3340 or TE 3340 or CE 4304 or EE 4304). (Same as CE 4337) (3-0) S
**CS 4341** Digital Logic and Computer Design (3 semester credit hours) Boolean algebra and logic circuits; synchronous sequential circuits; gate level design of ALSU, registers, and memory unit; register transfer operations; design of data path and control unit for a small computer; Input-Output interface. Credit cannot be received for both courses, (CS 4341 or TE 4341) and (CE 3320 or EE 3320). Prerequisites: (CE 2310 or EE 2310) or (CS 3340 or SE 3340 or TE 3340) and PHYS 2326. Corequisite: (CS 4141 or TE 4141). (Same as TE 4341) (3-0) S

**CS 4347** Database Systems (3 semester credit hours) This course emphasizes the concepts and structures necessary for the design and implementation of database management systems. Topics include data models, data normalization, data description languages, query facilities, file organization, index organization, file security, data integrity, and reliability. Prerequisite: CE 3345 or CS 3345 or SE 3345 or TE 3345. (Same as SE 4347) (3-0) S

**CS 4348** Operating Systems Concepts (3 semester credit hours) An introduction to fundamental concepts in operating systems: their design, implementation, and usage. Topics include process management, main memory management, virtual memory, I/O and device drivers, file systems, secondary storage management, and an introduction to critical sections and deadlocks. Prerequisites: (CS 3340 or SE 3340 or TE 3340 or equivalent) and (CS 3377 or SE 3377) and (CE 3345 or CS 3345 or SE 3345 or TE 3345). (Same as SE 4348) (3-0) S

**CS 4349** Advanced Algorithm Design and Analysis (3 semester credit hours) Asymptomatic analysis, recurrences, and graph algorithms. Algorithm design techniques such as greedy method, dynamic programming, and divide-and-conquer. Issues from computational complexity. Course emphasizes a theoretical approach. Prerequisites: CS 3305 with a grade of C or better, and (CE 3345 or CS 3345 or SE 3345 or TE 3345 or equivalent). (3-0) S

**CS 4352** Human Computer Interactions I (3 semester credit hours) Methods and principles of human-computer interaction (HCI), user-centered design (UCD), and usability evaluation. Provides broad overview of HCI and how HCI informs UCD processes throughout product development lifecycle. (Same as CGS 4352) (3-0) T

**CS 4353** Human Computer Interactions II (3 semester credit hours) Detailed exploration of human-computer interaction (HCI) through readings in journal articles and research reports. Practical experience in methodology typically used in the design of usable systems. (Same as CGS 4353) (3-0) T

**CS 4361** Computer Graphics (3 semester credit hours) Review of graphic display architecture and graphic input devices. Two- and three-dimensional transformations, matrix formulations, and concatenation. Clipping and windowing. Data structures for graphics systems, segmented display files, rings, etc. Hidden line and surface elimination. Shading. Graphics packages and applications. Prerequisites: MATH 2418, and (CE 2336 or CS 2336 or TE 2336), and (CE 3345 or CS 3345 or SE 3345 or TE 3345 or equivalent). (3-0) Y

**CS 4365** Artificial Intelligence (3 semester credit hours) Basic concepts and techniques that enable computers to perform intelligent tasks. Examples are taken from areas such as natural language understanding, computer vision, machine learning, search strategies and control, logic, and theorem proving. Prerequisite: CE 3345 or CS 3345 or SE 3345 or TE 3345 or equivalent. (3-0) Y

**CS 4371** Introduction to Big Data Management and Analytics (3 semester credit hours) This course focuses on scalable data management and mining algorithms for analyzing very large amounts of data (i.e., Big Data). Included topics are: Mapreduce, NoSQL systems (e.g., key-value stores, column-oriented data stores, stream processing systems), association rule mining, large scale supervised and unsupervised learning, and
applications including recommendation systems, web and big data security. Prerequisites: \textbf{CS 2336} and \textbf{CS 4347}. (3-0) Y

\textbf{CS 4372} Computational Methods for Data Scientists (3 semester credit hours) This course will focus on the application of computational tools to solve machine learning problems. Applicable languages may include Python, ‘R’, Weka, or others at the discretion of the instructor. Students will use these languages to apply machine learning concepts to problem data sets. Prerequisite: \textbf{CS 4375}. (3-0) Y

\textbf{CS 4375} Introduction to Machine Learning (3 semester credit hours) Algorithms for creating computer programs that can improve their performance through learning. Topics include: cross-validation, decision trees, neural nets, statistical tests, Bayesian learning, computational learning theory, instance-based learning, reinforcement learning, bagging, boosting, support vector machines, Hidden Markov Models, clustering, and semi-supervised and unsupervised learning techniques. Prerequisites: (\textbf{CS 3341} or \textbf{SE 3341}) and (\textbf{CE 3345} or \textbf{CS 3345} or \textbf{SE 3345} or \textbf{TE 3345} or equivalent). (3-0) Y

\textbf{CS 4376} Object-Oriented Design (3 semester credit hours) In-depth study of the features/advantages of object-oriented approach to problem solving. Special emphasis on issues of object-oriented analysis, design, implementation, and testing. Review of basic concepts of object-oriented technology (abstraction, inheritance, and polymorphism). Object-oriented programming languages, databases, and productivity tools. Prerequisite: (\textbf{CE 2336} or \textbf{CS 2336} or \textbf{TE 2336}) with a grade of C or better or equivalent. (Same as \textbf{SE 4376}) (3-0) S

\textbf{CS 4384} Automata Theory (3 semester credit hours) A review of the abstract notions encountered in machine computation. Topics include finite automata, regular expressions, PDAs, and context-free languages. Prerequisite: \textbf{CS 3305} with a grade of C or better. (3-0) S

\textbf{CS 4386} Compiler Design (3 semester credit hours) Basic phases of a compiler and their design principles. Topics include lexical analysis, basic parsing techniques such as LR(K) and LL(K) grammars. Prerequisite: \textbf{CE 3345} or \textbf{CS 3345} or \textbf{SE 3345} or \textbf{TE 3345} (or equivalent). (3-0) R

\textbf{CS 4389} Data and Applications Security (3 semester credit hours) Data as a critical resource. Threats to data and applications security including access control violations, integrity violations, unauthorized intrusions and sabotage; techniques to enforce security. Prerequisite: \textbf{CS 4347} or \textbf{SE 4347}. (3-0) Y

\textbf{CS 4390} Computer Networks (3 semester credit hours) The design and analysis of computer networks. Topics include the ISO reference model, transmission media, medium-access protocols, LANs, data link protocols, routing, congestion control, internetworking, and connection management. Credit cannot be received for both courses, (\textbf{CE 4390} or \textbf{CS 4390} or \textbf{TE 4390}) and \textbf{EE 4390}. Prerequisite: \textbf{CE 3345} or \textbf{CS 3345} or \textbf{SE 3345} or \textbf{TE 3345} or equivalent. (Same as \textbf{CE 4390} and \textbf{TE 4390}) (3-0) S

\textbf{CS 4391} Introduction to Computer Vision (3 semester credit hours) Techniques for manipulating and extracting information from digital images and video. Topics include color representations, analysis and processing based on image histograms, geometric transformations, convolutions, image blurring and sharpening, extraction of edges, matching, image and video motion. Prerequisites: \textbf{CE 3345} or \textbf{CS 3345} or \textbf{SE 3345} or \textbf{TE 3345} or equivalent. (3-0) Y

\textbf{CS 4392} Computer Animation (3 semester credit hours) Introduction to traditional animation. Kinematics of motion. Key framing. Coordinate systems and transformations (review), Euler angles and Quaternions, Catmull Rom and B-Splines, Advanced Key framing, articulated figures (forward kinematics), human and animal modeling (soft tissue, skin, etc.). Facial animation (parametric). Physically based modeling (rigid, collision detection). Physically based modeling (deformable). Behavioral and heuristic models. Algorithmic
animation. Optimization techniques. Animation languages and systems. Motion capture and real-time control. Virtual reality and animation. Rendering and temporal aliasing. 2D and 3D morphing. 3D modeling.

**Prerequisites:** MATH 2418 and (CE 3345 or CS 3345 or SE 3345 or TE 3345 or equivalent). (3-0) Y

**CS 4393** Computer and Network Security (3 semester credit hours) The study of security and vulnerabilities in computer and network systems. Common attacking techniques such as buffer overflow, viruses, worms, etc. Security in existing systems such as UNIX, Windows, and JVM. Fundamental access control and information flow concepts. Symmetric Ciphers such as DES and AES. Public-key encryption techniques and related number theory. Message authentication, hash functions, and digital signatures. Authentication applications, IP security and Web security. Prerequisite: CE 4348 or CS 4348 or SE 4348 or TE 4348 or equivalent. (3-0) Y

**CS 4394** Implementation of Modern Operating Systems (3 semester credit hours) This course focuses on developing systems implementation skills through a set of projects. Each project will explore one fundamental component of operating systems such as process scheduling, memory management, device drivers, file systems, and network communication management. The projects are expected to involve kernel-level programming. Prerequisites: (CE 4348 or CS 4348 or SE 4348 or TE 4348) and CS 3335, or equivalent programming experience. (3-0) R

**CS 4395** Human Language Technologies (3 semester credit hours) Introduction to human language technologies (HLT), the study of natural languages from a computational perspective. Topics include computational models of syntax and semantics, natural language applications (such as machine translation, speech processing, information retrieval, and information extraction), and general machine-learning techniques commonly used in state-of-the-art HLT research. Prerequisites: (CS 3341 or SE 3341) and (CE 3345 or CS 3345 or SE 3345 or TE 3345 or equivalent). (3-0) Y

**CS 4396** Networking Laboratory (3 semester credit hours) This course takes a lab-oriented approach to demonstrate how basic networking concepts are applied in a real network. The hands-on projects include setting up simple network topologies, configuring devices to run basic network protocols, and using various debugging tools to identify, locate, and fix common problems in networking. Prerequisite: CS 4390 or equivalent. (3-0) Y

**CS 4397** Embedded Computer Systems (3 semester credit hours) Introduction to embedded computer applications and concepts. Real-time operating systems and resource management. Real-time scheduling and communication. Senior data acquisition, processing and fusion. Error handling, fault tolerance, and graceful degradation. System performance analysis and optimization techniques. Includes a project to develop and analyze a small embedded computer application. Prerequisite: CE 4348 or CS 4348 or SE 4348 or TE 4348 or equivalent. (3-0) Y

**CS 4398** Digital Forensics (3 semester credit hours) Creating and preserving digital evidence, data recovery and evidence collection algorithms, evidence construction and reconstruction, methods for certifying evidence, storing evidence, data acquisition, forensic analysis algorithms, image files, network forensics, logging methods to trace back attacks and digital trails, e-mail investigations. Prerequisites: (CE 4348 or CS 4348 or SE 4348 or TE 4348) and (CE 4390 or CS 4390 or TE 4390) or equivalent. (3-0) Y

**CS 4399** Senior Honors in Computer Science (3 semester credit hours) For students conducting independent research for honors theses or projects. Topics may vary. Instructor consent required. (3-0) R

**CS 4475** Capstone Project (4 semester credit hours) This course is intended to provide hands-on experience in a data science project. Students will work in teams on projects and will be involved in formulating a
relevant problem, collecting the requisite data, finding a solution, and developing the necessary computational tools. The deliverables will include a final project report that details these steps and presentation of the project. Prerequisites: STAT 4355 and CS 4375. (Same as MATH 4475 and STAT 4475) (4-0) Y

**CS 4485** Computer Science Project (4 semester credit hours) This course is intended to complement theory and to provide an in-depth, hands-on experience in all aspects of a software development project. Students will work in teams on projects of interest to industry and will be involved in specifying the problem and its solution, designing and analyzing the solution, developing the software architecture, along with implementation and testing plans. The deliverables will include reports that document these steps as well as a final project report, including the challenges they faced, and a user manual of the developed system. Students will also make presentations as well as demonstrate their software. Additionally, this course will cover topics related to computer science profession including ethics and professional responsibility, entrepreneurship, leadership, and project management. Prerequisites: (CE 3345 or CS 3345 or SE 3345 or TE 3345), and (CE 3354 or CS 3354 or SE 3354 or equivalent), and at least three CS 43XX classes. (4-0) S

**CS 4V95** Undergraduate Topics in Computer Science (1-9 semester credit hours) Subject matter will vary from semester to semester. May be used as CS Guided Elective on CS degree plans. May be repeated for credit as topics vary (9 semester credit hours maximum). Prerequisite: (CE 3345 or CS 3345 or SE 3345 or TE 3345) and instructor consent required. ([1-9]-0) R

**CS 4V98** Undergraduate Research in Computer Science (1-9 semester credit hours) Topics will vary from semester to semester. May be repeated for credit as topics vary (9 semester credit hours maximum). Instructor consent required. ([1-9]-0) R