

Computer Science

Computing has become integrated into every aspect of business, science, engineering, medicine, and daily life, and is undergoing rapid evolution. NJIT's Department of Computer Science (<https://cs.njit.edu/>) is dedicated to meeting the demands of the private and public sectors by producing highly qualified computing professionals. We provide a top-notch computer science education that empowers our graduates to achieve remarkable success in their careers at renowned tech companies, start-ups, government agencies, and nonprofit organizations. Our programs equip students with both the theoretical foundations and practical skills necessary not only to excel in the short term, but also be adaptive to the ever-changing landscape of technology.

NJIT's Department of Computer Science is a thriving center for students at all levels looking to delve into the field of computing. The department offers B.A., B.S., M.S. and Ph.D. degree programs in computer science and evolving specialized interdisciplinary programs.

Undergraduate Degrees. The B.S. program provides a comprehensive and in-depth exploration of computer science, emphasizing technical expertise and advanced concepts. The B.A. program is designed for students with interests in liberal arts or management. Beyond traditional coursework, the department provides students with the Senior Capstone Program, a semester-long course that empowers students to apply their accumulated knowledge and skills to real-world projects, in collaboration with one of NJIT's 200 industry sponsors and partners.

Graduate Degrees. The MS in Computer Science offers a well-rounded advanced education in computer science. Students acquire theoretical knowledge and technical skills that enable them to pursue various types of careers in computing and technology. Students can shape their coursework based on their interests and professional needs, by selecting courses from various areas, including artificial intelligence, cybersecurity, computer algorithms, data management, programming languages and environments, systems, software engineering and other advanced topics. The department also offers cutting-edge specialized MS degrees in Cybersecurity and Privacy, in Software Engineering, and in Bioinformatics.

Ph.D. Degree. NJIT is positioned among the nation's most elite and productive research institutions, with an R1 (very high research activity) Carnegie classification. The Ph.D. degree offers to students the opportunity to engage, along with the department's faculty, in world-level research across a wide range of areas. The overarching objective of our research is to generate intellectual contributions and outcomes that will translate into tangible products and services, ultimately enhancing the lives of billions of people.

NJIT Faculty

A

Arafeh, Bassel, Senior University Lecturer

B

Basu Roy, Senjuti, Associate Professor

Borcea, Cristian M., Professor and Associate Dean for Strategic Initiatives

C

Calvin, James M., Professor

Curtmola, Reza, Professor

D

Dale, Mohit, University Lecturer

Ding, Xiaoning, Associate Professor

E

Eljabiri, Osama, Senior University Lecturer

Eren, Canan, Senior University Lecturer

Eskandari, Marzieh, Senior University Lecturer

G

Gerbessiotis, Alexandros, Associate Professor

Gotsman, Craig J., Distinguished Professor and Dean

H

Hamidli, Fuad, University Lecturer

Houle, Michael, Senior University Lecturer

I

Itani, Abdul-Rahman M., Senior University Lecturer

J

Jayarajah, Kasthuri, Assistant Professor

K

Kapleau, Jonathan, J., University Lecturer

Kellogg, Martin, Assistant Professor

Koutis, Ioannis, Associate Professor

Kunwar, Asif, Senior University Lecturer

L

Lay, Larry, Professor of Practice

Le, Huong, University Lecturer

Li, Jing, Assistant Professor

Li, Yajuan, University Lecturer

Liu, Chengjun, Professor

M

Mani, Kumar, Senior University Lecturer

Mili, Ali, Professor and Associate Dean for Academic Affairs

Musialski, Przemyslaw, Associate Professor

N

Naik, Kamlesh, University Lecturer

Nakayama, Marvin K., Professor

Nassimi, David, Associate Professor

Neamtiu, Iulian, Professor

Nicholson, Theodore L., Director of First Year Computer Science Education

O

Oria, Vincent, Professor

P

Perl, Yehoshua, Professor

Polyakov, Yuriy S., Research Professor

Q

Qerimaj, Jertishta, University Lecturer

R

Raza, Asad, Senior University Lecturer

Rohloff, Kurt, Associate Professor

Rusinkiewicz, Marek E., Dean Emeritus

Rutkowski, Wallace, Senior University Lecturer

S

Schieber, Baruch, Professor

Sharma, Shantanu, Assistant Professor

Shi, Cong, Assistant Professor

Shih, Frank Y., Professor

Sohn, Andrew, Associate Professor

Spirollari, Junilda, Senior University Lecturer

T

Theodoratos, Dimitrios, Associate Professor

V

Varadarajan, Ravi, Senior University Lecturer

W

Wang, Jason, T., Professor

Wang, Guiling (Grace), Distinguished Professor, Associate Dean for Research

Wei, Zhi, Professor

Wu, Chase Q., Professor

X

Xu, Pan, Assistant Professor

Y

Yao, Zhihao, Assistant Professor

Z

Zaidenberg, Ayelet, University Lecturer

Programs

- Computer Science - B.A. (<http://catalog.njit.edu/undergraduate/computing-sciences/computer-science/ba/>)
- Computer Science - B.S. (<http://catalog.njit.edu/undergraduate/computing-sciences/computer-science/bs/>)
- Computer and Business - B.S. (<http://catalog.njit.edu/undergraduate/computing-sciences/computer-science/computing-business-bs/>)

Double Majors (<http://catalog.njit.edu/undergraduate/academic-policies-procedures/special-degree-options/>)

- Computer Science and Applied Physics - B.S. (<http://catalog.njit.edu/undergraduate/computing-sciences/computer-science/cs-applied-physics-bs/>)
- Computer Science and Mathematical Sciences - B.S. (<http://catalog.njit.edu/undergraduate/computing-sciences/computer-science/cs-math-bs/>)
- Computer Science and Mathematical Sciences - Computational Mathematics - B.S. (<http://catalog.njit.edu/undergraduate/computing-sciences/computer-science/cs-math-bs-comp/>)

- Computer Science Minor (<http://catalog.njit.edu/undergraduate/computing-sciences/computer-science/minor/>) (not for Computer Engineering majors)
- Computer Science Minor (<http://catalog.njit.edu/undergraduate/computing-sciences/computer-science/minor-computer-engineering/>) (for Computer Engineering majors)

Computer Science Courses

BNFO 135. Programming for Bioinformatics. 3 credits, 3 contact hours (3;0;0).

The ability to use existing programs and to write small programs to access bioinformatics information or to combine and manipulate various existing bioinformatics programs has become a valuable part of the skill set of anyone working with biomolecular or genetic data. This course provides an understanding of the architecture of bioinformatics toolkits and experience in writing small bioinformatics programs using one or more of the scripting ("glue") languages frequently employed for such tasks.

BNFO 236. Programming for Bioinformatics II. 3 credits, 3 contact hours (3;0;0).

BNFO 330. Princ of Bioinformatics II. 3 credits, 3 contact hours (3;0;0).

BNFO 340. Data Analysis for Bioinformatics II. 3 credits, 3 contact hours (3;0;0).

Prerequisites: BNFO 240 and R120 101 or equivalent or permission of instructor. Advanced data analysis skills with applications to bioinformatics problems.

BNFO 482. Databases and Data Mining in Bioinformatics. 3 credits, 3 contact hours (3;0;0).

Prerequisites: BNFO 240 or equivalent or permission of instructor. Surveys biological databases and tools for managing them. Covers concepts and principles of data mining in bioinformatics. Hands-on experience for mining genomic data using ORACLE and SQL.

BNFO 488. Independent Study in Bioinformatics. 3 credits, 3 contact hours (0;0;3).

BNFO 491. Bioinformatics Senior Project. 3 credits, 3 contact hours (0;0;3).

Prerequisite: CS 490. Restriction: Senior standing in the Honors College and project proposal approval. A course similar to CS 491, with a project of greater depth and scope.

CS 100. Roadmap to Computing. 3 credits, 3 contact hours (3;0;0).

An introduction to programming and problem solving skills using Python or other very high level language. Topics include basic strategies for problem solving, constructs that control the flow of execution of a program and the use of high level data types such as lists, strings and dictionaries in problem representation. The course also presents an overview of selected topics in computing, such as networking and databases.

CS 101. Computer Programming and Problem Solving. 3 credits, 3 contact hours (3;0;0).

An introductory course that is designed for engineering freshmen. This course introduces students to the engineering problem solving process in the context of MATLAB. The emphasis is on the logical analysis of a problem and the formulation of a computer program leading to its solution. Topics include basic concepts of computer systems, algorithm design, programming languages and data abstraction. At the end of class, a comparison between MATLAB and C/C++ will be discussed to provide students a better understanding of the general concept of computer programming.

CS 103. Computer Science with Business Problems. 3 credits, 3 contact hours (3;0;0).

An introductory course in computer science, with applications to business and managerial decision making. Topics include basic concepts of computer systems, software engineering, algorithm design, programming languages and abstraction, with applications.

CS 104. Computer Programming and Graphics Problems. 3 credits, 3 contact hours (3;0;0).

An introductory course in computer science with applications in computer graphics for architecture. Emphasis on programming methodology using a high level language as the vehicle to illustrate the concepts. Topics include basic concepts of computer systems, software engineering, algorithm design, programming languages and data abstraction, with applications.

CS 106. Introduction to Computing. 3 credits, 3 contact hours (3;0;0).

An introduction to programming and problem solving skills for non-computing majors using Python programming languages. Topics include basic strategies for problem solving, constructs that control the flow execution of a program and the use of high level data types such as lists, strings, and dictionaries in problem representation. The course also presents an overview of selected "big idea" topics in computing.

CS 113. Introduction to Computer Science. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 100 or CS 103 with a grade C or better. Intensive introduction to computer science. Problem solving decomposition. Writing, debugging, and analyzing computer programs. Introduction to arrays and lists. Iteration and recursion. The Java language is introduced and used to highlight these concepts. A student receiving degree credit for CS 113 cannot receive degree credit for CS 115.

CS 114. Introduction to Computer Science II. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 113 with a grade C or better. A study of advanced programming topics with logical structures of data, their physical representation, and the design of computer algorithms operating on the structures. Course covers program specifications, correctness and efficiency, data abstraction, and algorithm analysis. Students receiving degree credit for CS 114 cannot receive degree credit for CS 116 or CS 505.

CS 115. Introduction to Computer Science in C++. 3 credits, 3 contact hours (3;0;0).

Fundamentals of computer science are introduced, with emphasis on programming methodology and problem solving. Topics include basic concepts of computer systems, software engineering, algorithm design, programming languages and data abstraction, with applications. The high level language C++ is fully discussed and serves as the vehicle to illustrate many of the concepts. CS majors should enroll in CS 113.

CS 116. Introduction to Computer Science II in C++.. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 115 with a grade C or better. A study of advanced programming topics with logical structures of data, their physical representation, design and analysis of computer algorithms operating on the structures, and techniques for program development and debugging. Course covers program specifications, correctness and efficiency, data abstraction, basic aspects of simple data structures, internal searching and sorting, recursion and string processing. Algorithmic analysis is also discussed. Students receiving degree credit for CS 116 cannot receive degree credit for CS 505 or CS 114.

CS 2. CS Elective. 3 credits, 3 contact hours (3;0;0).****CS 210. Technical History of Computing. 3 credits, 3 contact hours (3;0;0).**

Prerequisites: (CS 100 or CS 101 or CS 103 or CS 104 or CS 113 or CS 115 or BNFO 135) and any History and Humanities GER 200 level course and ENGL 101. This course is for students in computing majors. Students will gain a comprehensive overview of the evolution of computing from the start of recorded history through modern times. By studying history, you will understand the context of modern developments in CS/IT, including cyclical trends and why various approaches did or did not work. Learning where it all came from will also help young computer scientists to speak intelligently with older colleagues and managers in the workforce. Topics include mechanical calculating, analog computing, relay/tube computers, transistors, integrated circuits, I/O such as punch cards/paper tape/floppy disks, the minicomputer generation, the microcomputer revolution, development of graphical and network systems, early mobile computer, and modern history. A special focus on historic developments in New Jersey will be part of all lectures.

CS 241. Foundations of Computer Science I. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 114 and MATH 112 with a grade C or better. An introduction to the foundations of computer science with emphasis on the development of techniques for the design and proof of correctness of algorithms and the analysis of their computational complexity. Reasoning techniques based on propositional and predicate logic and relational calculus operations with applications to databases will also be introduced. Auxiliary topics such as combinatorics of finite sets, functions and relations, and graph-theory definitions and graph storage alternatives will also be examined.

CS 266. Game Modification Development. 3 credits, 3 contact hours (3;0;0).

Prerequisites: IT 102 or IT 114 or CS 114 or CS 116 with a grade C or better. This course introduces students to the basic concepts of game programming and development. Students will learn how to reprogram a professional game engine, or Modification (Mod) development as it is referred to in the industry. Students will work with C extensively. Students will work on their own game projects utilizing the professional game engine.

CS 276. 2D Game Development. 3 credits, 3 contact hours (3;0;0).

Prerequisites: (CS 265 and CS 266) or (IT 265 and IT 266) with a grade C or better. This course introduces students to the core concepts and skills necessary for the development of games utilizing 2D graphics. Students will learn how to set up and program their own 2D graphics based game engine. The engine will integrate 2D graphics, audio, input handling and network socket programming. Students will learn how to utilize their own custom 2D graphics and sounds into their projects. Once complete, students will have created two fully functional games.

CS 280. Programming Language Concepts. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 114 or CS 116 or IT 114 or equivalent with a grade C or better. Conceptual study of programming language syntax, semantics and implementation. Course covers language definition structure, data types and structures, control structures and data flow, run-time consideration, and interpretative languages.

CS 288. Intensive Programming in Linux. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 100 and CS 280 with a grade C or better. The course covers Linux programming with Apache Web and MySQL database using Php/Python and C as primary languages. It consists of four stages: basic tools such as Bash and C programming; searching trees and matrix computing, end-to-end applications such as one that constantly presents top 100 stocks; and extending the applications to run on multiple machines. The course provides students with hands-on experience for programming relatively large applications.

CS 3. CS Elective. 3 credits, 3 contact hours (3;0;0).****CS 301. Introduction to Data Science. 3 credits, 3 contact hours (3;0;0).**

Prerequisites: CS 114 and (MATH 333 or MATH 341) with a grade C or better. This course is designed for CS BS students to equip them with introductory principles as well as hands-on skills that are required to solve data science problems. The first part of the course focuses on learning models, formalism, and algorithmic techniques that are popular in data science and heavily used in practice. In the second part of the course, students are introduced to data science tools (e.g., Excel, Python).

CS 331. Database System Design & Mgmt. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 114 or CS 116 or IT 114 or equivalent with a grade C or better. Database system architecture; data modeling using the entity-relationship model; storage of databases; the hierarchical, network and relational data models; formal and commercial query languages; functional dependencies and normalization for relational database design; relation decomposition; concurrency control and transactions management. Student projects involve the use of a DBMS package.

CS 332. Principles of Operating Systems. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 114 or CS 116 or IT 114 or equivalent with a grade C or better. Organization of operating systems covering structure, process management and scheduling; interaction of concurrent processes; interrupts; I/O, device handling; memory and virtual memory management and file management.

CS 333. Introduction to UNIX Operating Systems. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 332 or equivalent and knowledge of C language. The course covers the UNIX system kernel including initialization, scheduling, context switching, process management, memory management, device management, and the file system. The course also includes the organization of shells, editors, utilities, and programming tools of the UNIX operating system.

CS 337. Performance Modeling in Computing. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 114 and (MATH 333 or MATH 341) with a grade C or better. Introduction to probability models and techniques useful in computer science. Performance evaluation, discrete-event simulation, classification and optimization are covered.

CS 341. Foundations of Computer Science II. 3 credits, 3 contact hours (3;0;0).

Prerequisites: (CS 241 or MATH 226) and CS 280 with a grade C or better. This course provides an introduction to automata theory, computability theory, and complexity theory. Theoretical models such as finite-state machines, push-down stack machines, and Turing machines are developed and related to issues in programming language theory. Also, the course covers undecidability and complexity classes P, NP, and NPC.

CS 345. Web Search. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 280 and CS 241 with a grade C or better. An introductory course on web searching. The architecture of a search engine. Information vs. data retrieval. Web crawling. Processing text (tokenization, stemming, stopwords, link analysis). The indexing process and inverted indexes. Query processing. Ranking algorithms based on indexes and links (e.g. Kleinberg's HITS, Google's PAGERANK). Retrieval Models. Search engine evaluation. Case studies (e.g. Google cluster architecture).

CS 350. Intro to Computer Systems. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 280 with a grade C or better. An introduction to the organization and architecture of computer systems, including the standard Von Neumann model and more recent architectural concepts. Among the topics covered are numeric data representation, assembly language organization, memory addressing, memory systems, both real and virtual, coding and compression, input/output structures treated as programmed, interrupt, and direct memory access, and functional organization of the CPU and the computer system.

CS 351. Introduction to Cybersecurity. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 241 and CS 356 with a grade C or better. This course will give a broad overview of cybersecurity. There are two main goals of this course. First, students will learn fundamental concepts of cybersecurity. Second, this course will help students gain knowledge of the applications to computer systems and communication security. Topics include basics of cryptography, access control, malware, software security, storage and file security, operating-system security, database security and secure communication protocols.

CS 356. Introduction to Computer Networks. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 280 with a grade C or better. This course provides an introduction to computer networks, with a special focus on Internet architecture and protocols. Topics include layered-network architectures, addressing, naming, forwarding, routing, communication reliability, the client-server model, web and email protocols. Besides the theoretical foundations, students acquire practical experience by programming reduced versions of real Internet protocols.

CS 357. Fundamentals of Network Security. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 356 or IT 420 with a grade C or better. This course is designed for Computer Science and Information Technology students. They must have a networking course before taking CS 357. IT students take IT 420 and Computer Science students take CS 356. This course offers an in-depth study of network security issues, types of computer and network attacks, and effective defenses. It provides both a theoretical foundation in the area of security and hands-on experience with various attack tools, firewalls, and intrusion-detection systems. Topics include: network scanning, TCP/IP stack fingerprinting, system vulnerability analysis, buffer overflows, password cracking, session hijacking, denial-of-service attacks, intrusion detection.

CS 366. 3D Game Development. 3 credits, 3 contact hours (3;0;0).

This course introduces students to the core concepts and skills necessary for the development of games utilizing 3D graphics. Students will learn how to set up and program their own 3D graphics based game engine using OpenGL. Students will learn how to load and display custom 3D models created using existing 3D modeling tools. Once complete, students will have created two fully functional 3D games and tools to work with them.

CS 370. Introduction to Artificial Intelligence. 3 credits, 4 contact hours (3;1;0).

Prerequisites: CS 114 and (MATH 226 or CS 241) with a grade C or better. An exploration of concepts, approaches and techniques of artificial intelligence. Emphasizes both underlying theory and applications. Topics include knowledge representation, parsing language, search, logic, abduction, uncertainty, and learning. LISP and Prolog programming languages are used extensively. Students are required to do programming assignments, complete a programming term project and review case studies.

CS 375. Introduction to Machine Learning. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 100 or CS 115 and MATH 333 or ECE 321 with a grade C or better. This is an introductory course to Machine Learning (ML). It consists of: (i) A smooth, example-based presentation of the fundamental notions of ML via simple algorithms and visualizable "toy" data sets. (ii) A tour of a selection of widely-used machine learning algorithms, including supervised, unsupervised, and reinforcement-based techniques, with applications on real data sets. The students are expected to implement basic algorithms and experiment with existing widely-used ML software libraries on real datasets. They will also gain exposure to the full development of an ML system via a course project.

CS 388. Android Application Development. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 288 with a grade C or better. This course introduces mobile application development for the Android platform. Students will learn skills necessary for creating and deploying applications with the Android Software Development Kit (SDK). The course is designed to introduce and familiarize students with programming in the Android environment. It starts with an examination of the basic components and concepts that define the Android platform, and then moves on to cover the specific structure that comprises an Android application. An overview of the most common tools and techniques for writing Android applications is included. The Android approach to user interfaces is described along with a discussion of some of the more common user-interface elements. Storage strategies for persistent information are also covered, including the use of the available SQLite Database features. The unique characteristics of programming for a mobile environment are introduced and explained. Hands on experience in the form of exercises and programming projects are included throughout the course to reinforce material that has been presented in lecture form.

CS 4.** CS Elective. 3 credits, 3 contact hours (3;0;0).**CS 408. Cryptography and Internet Security. 3 credits, 3 contact hours (3;0;0).**

Prerequisite: CS 351 with a grade C or better. Covers security requirements for telecommunication over the Internet and other communication networks, various conventional and public-key encryption protocols, digital encryption standard, RSA and ElGamal cryptographic systems, digital signature algorithm and analysis of its cryptosystem, and access-sharing schemes. Students receiving credit for CS 408 may not enroll in CS 608.

CS 433. Introduction to Linux Kernel Programming. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 288, CS 332, and CS 350. An introductory study of how the Linux operating system is built from scratch. As a hands-on course, students will perform intensive programming using the Linux kernel. The contents include booting, segmentation and paging, creating and destroying processes, process switching and scheduling, handling exceptions and interrupts, software interrupts, creating system calls, creating file systems, networking with TCP/IP, device driver writing and module programming. At the end of the course, students will be able to modify the Linux operating system to create their own.

CS 434. Advanced Database Systems. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 331 with a grade C or better. The course covers the basic concepts of traditional files and file processing, provides a "classic" introduction to the relational data model and its languages, and discusses database design methodology and application developments. Students are expected to learn the design of database application systems through a small project and to get some practical hands-on experience with commercial database management systems (DBMS) by writing application programs using the commercial DBMS query languages.

CS 435. Advanced Data Structures and Algorithm Design. 3 credits, 4 contact hours (3;1;0).

Prerequisites: CS 241 and CS 288 with a grade C or better. Advanced topics in data structures and algorithms, involving sequences, sets, and graphs such as searching, sorting, order statistics, balanced search tree operations, hash tables, graph traversals, graph connectivity and path problems. Algebraic and numeric algorithms. Performance measures, analysis techniques, and complexity of such algorithms.

CS 438. Interactive Computer Graphics. 3 credits, 3 contact hours (3;0;0).

Prerequisites: (CS 114 or CS 116) and MATH 337 with a grade C or better. This course introduces fundamental concepts of interactive graphics oriented toward computer-aided design systems. Such systems emerge in engineering, architecture, and manufacturing. Topics include computer data structures for representation of two- and three-dimensional objects and algorithms for definition, modification, and display of these objects in applications. This course will also discuss a selection of special topics in interactive graphics.

CS 439. Image Processing and Analysis. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 114 and MATH 333. This course is an intensive study of the fundamentals of image processing, analysis and understanding. Topics to be covered include: a brief review of the necessary mathematical tools, human visual perception, sampling and quantization, image transformation, enhancement, restoration, compression, reconstruction, image geometric transformation, matching, segmentation, feature extraction, representation and description, recognition and interpretation.

CS 440. Computer Vision. 3 credits, 3 contact hours (3;0;0).

Prerequisite: MATH 333. This course introduces basic concepts and methodologies of computer vision, and focuses on material that is fundamental and has a broad scope of applications. Topics include contemporary developments in all mainstream areas of computer vision e.g., Image Formation, Feature Representation, Classification and Recognition, Motion Analysis, Camera Calibration, Stereo Vision, Shape From X (shading, texture, motion, etc.), and typical applications such as Biometrics.

CS 441. Database Programming. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 114 or equivalent with a grade C or better. Many technologies have been developed due to the interplay between World-Wide Web development and databases on one hand and the growth of database applications in e-commerce on the other hand. Today, practically every e-commerce application has at least a Web component and a database component. Many languages have been developed in order to deal with these interactions. The course will focus on accessing databases through the Web but also cover new developments in the field.

CS 444. Big Data Systems. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 288 and CS 301 with a grade C or better. This course provides a broad coverage of topics on big data generation, transfer, storage, management, computing, and analytics with focus on state-of-the-art technologies and tools used in big data systems such as Hadoop. Real-life big-data applications and workflows in various domains are introduced as use cases to illustrate the development and execution of emerging big data-oriented solutions using HDFS, HBase, MapReduce/Spark, etc. deployed in cloud-based cluster environments.

CS 450. Data Visualization. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 288 and CS 301 with a grade C or better. The course provides students an introduction to computer graphics and the knowledge for designing, developing, and applying techniques for both information and volume visualization. Software tools such as Tableau and programming languages such as Python will be used to represent and interpret information in various visual forms, and volumetric visualization algorithms such as marching cubes and ray casting will be used for big data visualization of 3D datasets in scientific domains. Students will gain knowledge about theoretical design principles and apply them directly on real-world data, as part of assignments and course projects.

CS 458. Technologies-Network Security. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 351 with a grade C or better. This course provides both an in-depth theoretical study and a practical exposure to technologies that are critical in providing secure communication over the Internet. Topics include remote access security, web security, wireless security, e-mail security, spam and spam filtering techniques, computer viruses and internet worms, honeypots and honeynets, security liability issues and compliance.

CS 482. Data Mining. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 331 with a grade C or better. The course covers the concepts and principles of advanced data mining systems design; presents methods for association and dependency analysis, classification; prediction; and clustering analysis.

CS 485. Selected Topics In CS. 3 credits, 3 contact hours (3;0;0).

Restriction: junior standing and/or department approval. The study of new and/or advanced topics in an area of computer science not regularly covered in any other CS course. The precise topics to be covered in the course, along with prerequisites, will be announced in the semester prior to the offering of the course. A student may register for no more than two semesters of Special Topics.

CS 486. Topics in Computer Science/Information Systems. 3 credits, 3 contact hours (3;0;0).

Restriction: junior standing and/or department approval. A continuation of CS 485.

CS 488. Independent Study in Computer Science. 3 credits, 0 contact hours (0;0;0).

Restriction: Open only to Computer Science majors and who have the prior approval of the department and the CS faculty member who will guide the independent study. Independent studies, investigations, research, and reports on advanced topics in computer science. Students must prepare, in collaboration with their faculty mentor and in the semester prior to enrolling in this course, a detailed plan of topics and expected accomplishments for their independent study. This must have the approval of both the department and the faculty mentor. A student may register for no more than one semester of Independent Study.

CS 490. Guided Design in Software Engineering. 3 credits, 3 contact hours (3;0;0).

Prerequisites: CS 280 and CS 288 with a grade C or better. This course focuses on the methodology for developing software systems. Methods and techniques for functional requirements analysis and specifications, design, coding, testing and proving, integration and maintenance are discussed.

CS 491. Senior Project. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 490 with a grade C or better. Restriction: Senior standing and project proposal approval. An opportunity for the student to integrate the knowledge and skills gained in previous computer science work into a team-based project. The project involves investigation of current literature as well as computer implementation of either a part of a large program or the whole of a small system.

CS 492. Data Science Capstone I. 3 credits, 3 contact hours (3;0;0).

Restrictions: Senior standing. The Data Science (DS) Capstone Project spans two semesters and is intended to provide a real-world project-based learning experience for seniors in the BS DS program. The overall objectives of this course are to investigate the nature and techniques of a data-oriented computing development project. Projects are provided by faculty members or industry partners, or proposed by students who wish to become entrepreneurs. In DS Capstone I, teams of project participants will carry out market research, identify appropriate data science problems, collect and preprocess the needed data, define performance metrics, perform risk analysis, and finish an overall design of their solution that integrates various data analytics techniques. The course instructor will mentor and evaluate all projects in conjunction with an entrepreneurship board of industry, faculty, and alumni advisors.

CS 493. Data Science Capstone II. 3 credits, 3 contact hours (3;0;0).

Prerequisite: CS 492 with a grade C or better. The Data Science (DS) Capstone Project spans two semesters and is intended to provide a real-world project-based learning experience for seniors in the BS DS program. The overall objectives of this course are to investigate the nature and techniques of a data-oriented computing development project. Projects are provided by faculty members or industry partners, or proposed by students who wish to become entrepreneurs. In DS Capstone II, teams of project participants will refine their design, implement and integrate component techniques into a complete software solution, present data analysis results, evaluate the system performance, and validate the proposed solution. The course instructor will mentor and evaluate all projects in conjunction with an entrepreneurship board of industry, faculty, and alumni advisors.