

Computer Science & Engineering

Phone: 314-935-6160
 Website: <https://cse.wustl.edu/academics/undergraduate/index.html>

Courses

Prerequisites are advisory in our course listings, but students are cautioned against taking a course without the necessary background. Note that if one course mentions another as its prerequisite, the prerequisites of the latter course are implied to be prerequisites of the former course as well. Students in doubt of possessing the necessary background for a course should correspond with the course's instructor.

CSE 1007 Data Science Playground

Data science plays an increasingly important role in research, industry, and government. Researchers seek to understand behavior and mechanisms, companies seek to increase profits, and government agencies make policies intended to improve society. While we are awash in an abundance of data, making sense of data is not always straightforward. At its core, students of data science learn techniques for analyzing, visualizing, and understanding data. However, students must also cultivate curiosity about data, including the data's provenance, ethical considerations such as bias, and skepticism concerning correlation and causality. In the beginning, students investigate a curated collection of data sets, asking questions they find interesting and exploring data using a popular platform for such studies. Thereafter, researchers on campus present their work in the context of data science, challenging students to explore data in the domain of their research areas. Throughout the course, students present their findings in their group and to the class. This course is offered in an active-learning setting in which students work in small teams. The course has no prerequisites, and programming experience is neither expected nor required. All credit for this pass/fail course is based on work performed in the scheduled class time.

Credit 1 unit.

Typical periods offered: Fall

CSE 1301 Introduction to Computer Science

An introduction to software concepts and implementation, emphasizing problem solving through abstraction and decomposition. Introduces processes and algorithms, procedural abstraction, data abstraction, encapsulation, and object-oriented programming. Recursion, iteration, and simple data structures are covered. Concepts and skills are mastered through programming projects, many of which employ graphics to enhance conceptual understanding. Java, an object-oriented programming language, is the vehicle of exploration. Active-learning sessions are conducted in a studio setting in which students interact with each other and the professor to solve problems collaboratively. Prerequisites: Comfort with algebra and geometry at the high school level is assumed. Patience, good planning, and organization will promote success. This course assumes no prior experience with programming. *CSE 1301 has mandatory evening exams; the specific days and times will be listed under assessments on the course section once confirmed.

Credit 3 units. BU: SCI

Typical periods offered: Fall, Spring, Summer

CSE 1302 Introduction to Computer Engineering

This course introduces students to fundamental concepts in the basic operation of computers, ranging from desktops and servers to microcontrollers and handheld devices. Subjects include digital and analog input/output, sensing the physical world, information representation, basic computer architecture and machine language, time-critical computation, machine-to-machine communication and protocol design. Students will use both desktop systems and handheld microcontrollers for laboratory experiments. Active-learning sessions are conducted in a studio setting in which students interact with each other and the professor to solve problems collaboratively. Prerequisite: CSE 131.

Credit 3 units. BU: SCI

Typical periods offered: Fall, Spring, Summer

CSE 1391 Seminar: Computer Science I

A seminar and discussion session that complements the material studied in CSE131. Provides background and breadth for the disciplines of computer science and computer engineering. Features guest lectures and highly interactive discussions of diverse computer science topics. Highly recommended for majors and for any student seeking a broader view of computer science or computer engineering. Pass/Fail only.

Credit 1 unit.

Typical periods offered: Fall, Spring

CSE 1392 Seminar: Computer Science II

This course is a seminar and discussion session that complements the material studied in CSE 132. It provides background and breadth for the disciplines of computer science and computer engineering, and it features guest lectures and highly interactive discussions of diverse computer science topics. Highly recommended for majors and for any student seeking a broader view of computer science or computer engineering.

Credit 1 unit.

Typical periods offered: Spring

CSE 1997 Computer Science Department Seminar

This seminar will host faculty, alumni, and professionals to discuss topics related to the study and practice of computer science. Attendance is mandatory to receive a passing grade.

Credit 0 units.

Typical periods offered: Fall, Spring

CSE 2000 Engineering and Scientific Computing

This course provides an introduction to numerical methods for scientific computation which are relevant to engineering problems. Topics addressed include interpolation, integration, linear systems, least-squares fitting, nonlinear equations and optimization and initial value problems. Basic procedural programming concepts (procedural and data abstraction, iteration, recursion) will be covered using MATLAB. C will be briefly covered so the students understand that the algorithms and programming concepts apply in both. Prerequisite: Math 132.

Credit 3 units.

CSE 2004 Web Development

This course explores elementary principles for designing, creating, and publishing effective websites and web application front-ends. Topics include page layout concepts, design principles, HTML, CSS, JavaScript, front-end frameworks like Angular and React, and other development tools. Students apply the topics by creating a series of websites that are judged based on their design and implementation. Prerequisite: CSE 131 or equivalent experience.

Credit 3 units. BU: SCI
Typical periods offered: Fall, Spring

CSE 2107 Introduction to Data Science

This course provides an introduction to data science and machine learning, and it focuses on the practical application of models to real-world supervised and unsupervised learning problems. We will discuss methods for linear regression, classification, and clustering and apply them to perform sentiment analysis, implement a recommendation system, and perform image classification or gesture recognition. One of the main objectives of the course is to become familiar with the data science workflow, from posing a problem to understanding and preparing the data, training and evaluating a model, and then presenting and interpreting the results. We will also touch on concepts such as similarity-based learning, feature engineering, data manipulation, and visualization. The course uses Python, which is currently the most popular programming language for data science. This course does not teach programming in Python. Prerequisites: CSE 131, MATH 233, and CSE 247 (can be taken concurrently). Credit 3 units.

Typical periods offered: Fall, Spring

CSE 2201 Internet of Things

With the advent of the Internet of Things, we can address, control, and interconnect formerly isolated objects to create new and interesting applications. In this course we study fundamental technologies behind Internet-of-Things devices, and Accessories, which include smart watches, health monitors, toys, and appliances. In addition to learning about IoT, students gain hands-on experience developing multi-platform solutions that control and communicate with Things using via mobile device friendly interfaces. Students apply their knowledge and skill to develop a project of their choosing using topics from the course. Prerequisite: CSE 132.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 2301 Introduction to Parallel and Concurrent Programming

This course explores concepts, techniques, and design approaches for parallel and concurrent programming. We study how to write programs that make use of multiple processors for responsiveness and that share resources reliably and fairly. Parallel programming concepts include task-level, functional, and loop-level parallelism. Concurrent programming concepts include threads, synchronization, and locks. We cover how to adapt algorithms to achieve determinism and avoid data races and deadlock. Concepts and skills are acquired through the design and implementation of software projects. Prerequisite: CSE 131.

Credit 3 units.
Typical periods offered: Fall, Spring

CSE 2302 Programming Skills Workshop

This course provides an overview of practical implementation skills. Topics include compilation and linking, memory management, pointers and references, using code libraries, testing and debugging. Prerequisite: CSE 247.

Credit 1 unit.

Typical periods offered: Fall

CSE 2307 Programming Tools and Techniques

This course offers an introduction to the tools and techniques that allow programmers to write code effectively. Tools covered include version control, the command line, debuggers, compilers, unit testing, IDEs, bug trackers, and more. The course also places a heavy emphasis on code quality: how can we write code that is functional and that also meets quality standards? Peer review exercises will be used to show the importance of code craftsmanship. Prerequisite: CSE 131.

Credit 3 units.

Typical periods offered: Spring

CSE 2400 Logic and Discrete Mathematics

Introduces elements of logic and discrete mathematics that allow reasoning about computational structures and processes. Generally, the areas of discrete structures, proof techniques, probability and computational models are covered. Topics typically include propositional and predicate logic; sets, relations, functions and graphs; direct and indirect proof methods, induction and recursion; finite state machines and regular languages. Prerequisite: CSE 131 Note that this course will be held in-person. *CSE 2400 has mandatory evening exams; the specific days and times will be listed under assessments on the course section once confirmed.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 2402 Discrete Math Seminar

This semester, this course will focus on clear, compelling, and concise proofs. Students will read, critique, and write proofs throughout the semester, with the goal of becoming more proficient in understanding, authoring, and evaluating proofs. Admission is by permission of the instructor, and this course carries elective credit for all computer science programs. Students will be admitted to this course based on their performance in other math and computer science courses, and with the understanding that they will serve as TAs for our proof-based classes next semester.

Credit 3 units.

Typical periods offered: Fall

CSE 2407 Data Structures and Algorithms

Study of fundamental algorithms, data structures, and their effective use in a variety of applications. Emphasizes importance of data structure choice and implementation for obtaining the most efficient algorithm for solving a given problem. A key component of this course is worst-case asymptotic analysis, which provides a quick and simple method for determining the scalability and effectiveness of an algorithm. Online textbook purchase required. Prerequisite: CSE 131/501N, and fluency with summations, derivatives, and proofs by induction. *CSE 2407 has mandatory evening exams; the specific days and times will be listed under assessments on the course section once confirmed.

Credit 3 units.

Typical periods offered: Fall, Spring, Summer

CSE 2497 Seminar: Data Structures and Algorithms

The content of this seminar will vary by semester, but it will generally complement the material taught in CSE 247 Data Structures and Algorithms. Corequisite: CSE 247.

Credit 1 unit.

Typical periods offered: Fall, Spring

CSE 2506 Introduction to Human-Centered Design

This course provides an introduction to human-centered design through a series of small user interface development projects covering information foraging, learnability, and problem solving. Projects will begin with reviewing a relevant model of human behavior. Each project will then provide an opportunity to explore how to apply that model in the design of a new user interface. Prerequisite: CSE 131.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 2600 Introduction to Digital Logic and Computer Design

Introduction to design methods for digital logic and fundamentals of computer architecture. Boolean algebra and logic minimization techniques; sources of delay in combinational circuits and effect on circuit performance; survey of common combinational circuit components; sequential circuit design and analysis; timing analysis of sequential circuits; use of computer-aided design tools for digital logic design (schematic capture, hardware description languages, simulation); design of simple processors and memory subsystems; program execution in simple processors; basic techniques for enhancing processor performance; configurable logic devices. Prerequisites: CSE 131

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 3050 Responsible Data Science

In today's data-driven world, understanding the ethical and responsible use of data is critical. This course provides a comprehensive overview of the key principles and challenges in responsible data science. The first wave of data science focuses on accuracy and efficiency-what we can do with data. The second wave focuses on responsibility-what we should or should not do. This technical course addresses ethical and responsible data science, including legal compliance, data quality, algorithmic fairness and diversity, transparency, privacy, and data protection. Through case studies, hands-on activities, and discussions, students will learn how to apply ethical frameworks and data governance principles to real-world data science problems. By the end of the course, participants will be equipped with the tools to make informed, ethical decisions in data science practices.

Credit 3 units.

Typical periods offered: Spring

CSE 3101 Introduction to Intelligent Agents Using Science Fiction

The course uses science-fiction short stories, TV episodes, and movies to motivate and introduce fundamental principles and techniques in intelligent agent systems. Topics covered may include game theory, decision theory, machine learning, distributed algorithms, and ethics. Prerequisite: CSE 247.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 3103 Artificial Intelligence Lab

Project course in artificial intelligence. Students are required to solve AI problems for which there are not existing canonical approaches. Standard approaches are discussed, but the objective is to be creative and to produce working software rather than to master concepts. Projects usually contain elements of computer vision, natural language processing, game playing, heuristic search and heuristic optimization, the application and representation of knowledge and automatic inference. Prerequisites: CSE 132 and 240, familiarity with UNIX.

Credit 3 units.

Typical periods offered: Fall

CSE 3104 Data Manipulation and Management

As the base of data science, data needs to be acquired, integrated and preprocessed. This important step in the data science workflow ensures both quantity and quality of data and improves the effectiveness of the following steps of data processing. Students will gain an understanding of concepts and approaches of data acquisition and governance including data shaping, information extraction, information integration, data reduction and compression, data transformation as well as data cleaning. The course will further highlight the ethical responsibility of protecting the integrity of data and proper use of data. Prerequisites: CSE 131, CSE 217A; Corequisite: CSE 247

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 3106 Social Network Analysis

This course looks at social networks and markets through the eyes of a computer scientist. We will look at questions including, Why are acquaintances rather than friends more likely to get us job opportunities? and, Why do the rich get richer? We begin by studying graph theory (allowing us to study the structure) and game theory (allowing us to study the interactions) of social networks and market behavior at the introductory level. Among other topics, we will study auctions, epidemics, and the structure of the Internet (including web searches). This course examines the intersection of computer science, economics, sociology, and applied mathematics. Prerequisites: CSE 240 and CSE 247

Credit 3 units.

Typical periods offered: Fall

CSE 3300 Rapid Prototype Development and Creative Programming

This course uses web development as a vehicle for developing skills in rapid prototyping. Students acquire the skills to build a Linux web server in Apache, to write a web site from scratch in PHP, to run an SQL database, to perform scripting in Python, to employ various web frameworks, and to develop modern web applications in client-side and server-side JavaScript. The course culminates with a creative project in which students are able to synthesize the course material into a project of their own interest. The course implements an interactive studio format: after a formal presentation of a topic, students develop a related project under the supervision of the instructor. Prerequisite: CSE 131 Revised: 2019-02-21

Credit 3 units.

Typical periods offered: Fall, Spring, Summer

CSE 3302 Object-Oriented Software Development Laboratory

Intensive focus on practical aspects of designing, implementing and debugging software, using object-oriented, procedural, and generic programming techniques. The course emphasizes familiarity and proficiency with a wide range of C++ language features through hands-on practice completing studio exercises and lab assignments, supplemented with readings and summary presentations for each session. Prerequisites: CSE 131 and CSE 247 *CSE 3302 has mandatory evening exams; the specific days and times will be listed under assessments on the course section once confirmed.

Credit 3 units.

Typical periods offered: Fall, Spring, Summer

CSE 3401 Parallel and Sequential Algorithms

The course aims to teach students how to design, analyze and implement parallel algorithms. The emphasis is on teaching fundamental principles and design techniques that easily transfer over to parallel programming. These techniques include divide and conquer, contraction, the greedy method, and so on. Prerequisites: CSE 240, CSE 247, and Math 310.

Credit 3 units.

CSE 3407 Analysis of Algorithms

This course introduces techniques for the mathematical analysis of algorithms, including randomized algorithms and non-worst-case analyses such as amortized and competitive analysis. It also introduces the standard paradigms of divide-and-conquer, greedy, and dynamic programming algorithms, as well as reductions, and it provides an introduction to the study of intractability and techniques to determine when good algorithms cannot be designed. Prerequisites: (CSE 240 or Math 310) and CSE 247 *CSE 3407 has mandatory evening exams; the specific days and times will be listed under assessments on the course section once confirmed.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 3487 Analysis of Algorithms Recitation

This course is the recitation component of CSE 347. Registration and attendance for 347R is mandatory for students enrolled in 347.

Credit 0 units.

Typical periods offered: Fall

CSE 3601 Introduction to Systems Software

This course is an introduction to the hardware and software foundations of computer processing systems. The course provides a programmer's perspective of how computer systems execute programs and store information. The course material aims to enable students to become more effective programmers, especially when dealing with issues of performance, portability and robustness. It also serves as a foundation for other system courses (e.g., those involving compilers, networks, and operating systems), where a deeper understanding of systems-level issues is required. Topics covered include machine-level code and its generation by optimizing compilers, performance evaluation and optimization, computer arithmetic, memory organization and management, and supporting concurrent computation. Prerequisites: CSE 131 and CSE 132.

Credit 3 units.

Typical periods offered: Fall, Spring, Summer

CSE 3602 Computer Architecture

This course explores the interaction and design philosophy of hardware and software for digital computer systems. Topics include: Processor architecture, Instruction Set Architecture, Assembly Language, memory hierarchy design, I/O considerations, and a comparison of computer architectures. Prerequisite: CSE 260M.

Credit 3 units.

Typical periods offered: Fall

CSE 3605 Elements of Computing Systems

All computers are made up of 0s and 1s. However, the conceptual gap between the 0s and 1s and the day to day operation of modern computers is enormously wide. This fast-paced course aims to bridge the divide by starting with simple logic gates and building up the levels of abstraction until one can create games like Tetris. There is no specific programming language requirement, but some experience with programming is needed. Prerequisite: CSE 247

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4000 Independent Study

This course allows the student to investigate a topic in computer science and engineering of mutual interest to the student and a mentor. Interested students are encouraged to approach and engage faculty to develop a topic of interest. A form (<https://wustl.box.com/s/gp2mffz8r7ql9rgsil5zfyz5pz05b8tb>) declaring the agreement must be filed in the departmental office. This course carries university credit, but it does not count toward a CSE major or minor. To arrange for CSE major or minor credit for independent study, a student must enroll in CSE 400E instead of CSE 400. Consult also CSE 400E.

Credit 6 units.

Typical periods offered: Fall, Spring, Summer

CSE 4001 Independent Study

Investigation of a topic in computer science and engineering of mutual interest to the student and a mentor. Interested students are encouraged to approach and engage faculty to develop a topic of interest. A form (<https://wustl.box.com/s/gp2mffz8r7ql9rgsil5zfyz5pz05b8tb>) declaring the agreement must be filed in the departmental office. Credit earned for CSE400E can be counted toward a student's major or minor program, with the consent of the student's advisor. See also: CSE400

Credit 6 units.

Typical periods offered: Fall, Spring, Summer

CSE 4101 AI and Society

AI has made increasing inroads in a broad array of applications, many that have socially significant implications. This course will study a number of such applications, focusing on issues such as AI used for social good, fairness and accountability of AI, and potential security implications of AI systems. Prerequisite: CSE 247.

Credit 3 units.

Typical periods offered: Fall

CSE 4102 Introduction to Artificial Intelligence

The discipline of artificial intelligence (AI) is concerned with building systems that think and act like humans or rationally on some absolute scale. This course is an introduction to the field, with special emphasis on sound modern methods. The topics include knowledge representation, problem solving via search, game playing, logical and probabilistic reasoning, planning, dynamic programming, and reinforcement learning. Programming exercises concretize the key methods. The course targets graduate students and advanced undergraduates. Evaluation is based on written and programming assignments, a midterm exam and a final exam. Prerequisites: CSE 247, ESE 326 (or Math 3200 or Math 3211 or Engr 328), and Math 233. This course will be taught using Zoom and will be recorded. *CSE 4102 has mandatory evening exams; the specific days and times will be listed under assessments on the course section once confirmed.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4103 Web Privacy and Security

This course introduces the fundamental concepts of web privacy and security. It explores the challenges and techniques related to protecting personal privacy online and securing web applications from threats and vulnerabilities. Why take this course: Given that the vast majority of user-facing software is consumed through the web platform, developing an understanding of the web ecosystem is crucial in improving user privacy and security, overall. Topics include: Webpage execution (e.g., JavaScript execution), Browser security model (e.g., site isolation), Online tracking (e.g., cookies, fingerprinting), Privacy enhancing technologies (e.g., tracker blocking), Technology policy (privacy laws, policies). Prereqs: CSE 247

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4106 Data Science for Complex Networks

This course examines complex systems through the eyes of a computer scientist. We will use the representative power of graphs to model networks of social, technological, or biological interactions. Network analysis provides many computational, algorithmic, and modeling challenges. We begin by studying graph theory, allowing us to quantify the structure and interactions of social and other networks. We will then explore how to practically analyze network data and how to reason about it through mathematical models of network structure and evolution. We will also investigate algorithms that extract basic properties of networks in order to find communities and infer node properties. Finally, we will study a range of applications including robustness and fragility of networks such as the internet, spreading processes used to study epidemiology or viral marketing, and the ranking of webpages based on the structure of the webgraph. This course combines concepts from computer science and applied mathematics to study networked systems using data mining. Prerequisites: CSE 247, ESE 326, MATH 309, and programming experience (note: we will parse data and analyze networks using Python)

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4107 Introduction to Machine Learning

The field of machine learning is concerned with the question of how to construct computer programs that automatically improve with experience. This course is a broad introduction to machine learning, covering the foundations of supervised learning and important supervised learning algorithms. Topics to be covered are the theory of generalization (including VC-dimension, the bias-variance tradeoff, validation, and regularization) and linear and non-linear learning models (including linear and logistic regression, decision trees, ensemble methods, neural networks, nearest-neighbor methods, and support vector machines). Prerequisites: CSE 247, ESE 326 (or Engr 328), Math 233, and Math 309 (can be taken concurrently).

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4109 Introduction to AI for Health

Artificial Intelligence (AI) has emerged as a powerful tool for solving complex health problems using advanced data-driven techniques. AI for health is fueled by both the advancement in AI methods and the increasing availability of health data. This course will introduce the interdisciplinary field of AI for health including applications of AI in various healthcare domains. Students will read and critique research papers and perform projects to develop AI models and solutions for health care.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4202 Operating Systems Organization

This course involves a hands-on exploration of core OS abstractions, mechanisms and policies in the context of the Linux kernel. Readings, lecture material, studio exercises, and lab assignments are closely integrated in an active-learning environment in which students gain experience and proficiency writing OS code, as well as tracing and evaluating OS operations via user-level programs and kernel-level monitoring tools. Topics include: system calls, interrupt handling, kernel modules, concurrency and synchronization, proportional and priority-based scheduling of processes and threads, I/O facilities, memory management, virtual memory, device management, and file system organization. Prerequisite: CSE 361S

Credit 3 units.

Typical periods offered: Fall, Spring, Summer

CSE 4205 Programming Systems and Languages

A systematic study of the principles, concepts and mechanisms of computer programming languages: their syntax, semantics and pragmatics; the processing and interpretation of computer programs; programming paradigms; and language design. Illustrative examples are selected from a variety of programming language paradigms.

Prerequisites: CSE 332S

Credit 3 units.

Typical periods offered: Fall, Spring, Summer

CSE 4207 Cloud Computing With Big Data Applications

This course provides an overview of the tools necessary to harness big data on the cloud for real world analytic applications. Each lecture will cover an important cloud computing concept or framework, and will be accompanied by a lab. There will be an emphasis on hands-on experience through using each of the tools taught in this course in a small project. Topics include Cloud based security and storage, Linux, Docker and Kubernetes, Data modeling through JSON and SQL, Database concepts and storage architectures, distributed systems, and finally real-world applications. Labs will build on each other and require the completion of the previous week's lab. The course will end with a multi-week, open-ended final project. Prerequisites: CSE 131, CSE 247, and CSE 330.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4208 Multi-Paradigm Programming in C++

Intensive focus on how modern C++ language features support procedural, functional, generic, and object-oriented programming paradigms and allow those paradigms to be applied both separately and in combination. Topics covered will include various C++ language features and semantics, especially from the C++11 standard onward, with studio exercises and lab assignments designed to build proficiency in using them effectively within and across the different programming paradigms. Prerequisite: CSE 332S or CSE 504N; or graduate standing and basic proficiency in C++.

Credit 3 units.

Typical periods offered: Fall

CSE 4301 Translation of Computer Languages

The theory of language recognition and translation is introduced in support of compiler construction for modern programming languages. Topics include syntactic and semantic analysis, symbol table management, code generation, and runtime libraries. A variety of parsing methods is covered, including top-down and bottom-up. Machine problems culminate in the course project, for which students construct a working compiler.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4303 Introduction to Computer Security

Secure computing requires the secure design, implementation, and use of systems and algorithms across many areas of computer science. Fundamentals of secure computing such as trust models and cryptography will lay the groundwork for studying key topics in the security of systems, networking, web design, machine learning algorithms, mobile applications, and physical devices. Human factors, privacy, and the law will also be considered. Hands-on practice exploring vulnerabilities and defenses using Linux, C, and Python in studios and lab assignments is a key component of the course.

Prerequisites: CSE 247 and either CSE 361 or CSE 332.

Credit 3 units.

Typical periods offered: Fall, Spring, Summer

CSE 4304 Reverse Engineering and Malware Analysis

This course will focus on reverse engineering and malware analysis techniques. Reverse engineering -- the process of deconstructing an object to reveal its design and architecture -- is an essential skill in the information security community. Although hackers often use reverse engineering tools to discover and exploit vulnerabilities, security analysts and researchers must use reverse engineering techniques to find what a specific malware does, how it does it, and how it got into the system. In this course, we will explore reverse engineering techniques and tools, focusing on malware analysis. Students will use and write software during in-class studios and homework assignments to illustrate mastery of the material. Prerequisite: CSE 361S.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4305 Database Management Systems

A study of data models and the database management systems that support these data models. The design theory for databases is developed and various tools are utilized to apply the theory. General query languages are studied and techniques for query optimization are investigated. Integrity and security requirements are studied in the context of concurrent operations on a database, where the database may be distributed over one or more locations. The unique requirements for engineering design databases, image databases, and long transaction systems are analyzed. Prerequisite: CSE 247.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4307 Software Engineering Workshop

An introduction and exploration of concepts and issues related to large-scale software systems development. Areas of exploration include technical complexities, organization issues, and communication techniques for large-scale development. Students participate through teams emulating industrial development. The projects cover the principal system development life-cycle phases from requirements analysis, to software design, and to final implementation. Issues relating to real-time control systems, human factors, reliability, performance, operating costs, maintainability and others are addressed and resolved in a reasonable manner. Book Requirement: <https://www.oreilly.com/library/view/kubernetes-up-and/9781492046523/>. Prerequisite: Junior or Senior standing, and CSE 330S.

Credit 3 units.

Typical periods offered: Fall, Spring, Summer

CSE 4308 Mobile Application Development

Learn how to create iOS apps in the Swift programming language. This course covers a variety of topics in the development of modern mobile applications, with a focus on hands-on projects. Students will create multiple fully-functional apps from scratch. The course emphasizes object-oriented design patterns and real-world development techniques. Time is provided at the end of the course for students to work on a project of their own interest. Prerequisite: CSE 247.

Credit 3 units.

Typical periods offered: Fall, Spring, Summer

CSE 4309 Mobile Application Development II

In this course, students will work in groups to design, develop, test, publish, and market an iOS mobile application. Throughout the semester, students will operate in different roles on a team, serving as lead developer, tester, and project manager. Topics include how to publish a mobile application on an app store, APIs and tools for testing

and debugging, and popular cloud-based SDKs used by developers. This course requires completion of the iOS version of CSE 438 Mobile Application Development or the appropriate background knowledge of the iOS platform.

Credit 3 units.

Typical periods offered: Spring

CSE 4393 Seminar: Capture the Flag (CTF) Studio

This course provides a collaborative studio space for hands-on practice solving security-relevant puzzles in Capture The Flag (CTF) format. Students will engage CTF challenges individually and in teams, and online CTF resources requiring (free) account signup may be used. Students will be encouraged to attempt challenges commensurate with their ability, but no prior CTF experience or security knowledge is assumed. The goal of the course is to build skills in the fundamentals of security analysis, including usage of the Linux command line and console-based security tools, creativity in applying theoretical knowledge to practical challenges, and confidence in approaching under-specified problems. Prerequisite/corequisite: CSE 433S or equivalent.

Credit 1 unit.

Typical periods offered: Fall, Spring

CSE 4397 Technical Interviewing

Students will study, give, and receive technical interviews in this seminar course. The focus will be on improving student performance in a technical interview setting, with the goal of making our students as comfortable and agile as possible with technical interviews. Professionals from the local and extended Washington University community will mentor our students in this seminar.

Credit 2 units.

Typical periods offered: Spring

CSE 4402 Introduction to Cryptography

This course is an introduction to modern cryptography, with an emphasis on its theoretical foundations. Topics will include one-way functions, pseudorandom generators, public key encryption, digital signatures, and zero-knowledge proofs. Prerequisites: CSE 240 and CSE 247.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4470 Introduction to Formal Languages and Automata

An introduction to the theory of computation, with emphasis on the relationship between formal models of computation and the computational problems solvable by those models. Specifically, this course covers finite automata and regular languages; Turing machines and computability; and basic measures of computational complexity and the corresponding complexity classes. Prerequisites: CSE 240 and CSE 247

Credit 3 units.

Typical periods offered: Spring

CSE 4500 Video Game Programming

This course teaches the core aspects of a video game developer's toolkit. Students work in groups and with a large game software engine to create and playtest a full-featured video game. Students will explore topics around the design of games through analysis of current games. Students have the opportunity to explore additional topics including graphics, artificial intelligence, networking, physics, and user interface design through their game project. Prerequisite: CSE247.

Credit 3 units.

Typical periods offered: Fall, Spring, Summer

CSE 4501 Video Game Programming II

This course is a continuation of CSE 450A Video Game Programming I. Students will work in groups and with a large game software engine to make a full-featured video game. Students will have the opportunity to work on topics in graphics, artificial intelligence, networking, physics, user interface design, and other topics. Prerequisites: CSE 450A and permission of instructor.

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4502 Computer Graphics

Introduction to computer graphics. The course covers fundamental concepts, data structures, and algorithms related to the construction, display, and manipulation of three-dimensional objects. Topics include scan-conversion, basic image processing, transformations, scene graphs, camera projections, local and global rendering, fractals, and parametric curves and surfaces. Students develop interactive graphics programs using C++ language. Prerequisite: CSE 332S and Math 309

Credit 3 units.

CSE 4504 Software Engineering for External Clients

Teams of students will design and develop a solution to a challenging problem posed by a real-world client. Students will gain experience with a variety of facets of software development, such as gathering and interpreting requirements, software design/architecture, UI/UX, testing, documentation, and developer/client interactions. Prerequisite: E81 CSE 330S or E81 CSE 332S and at least junior standing

Credit 3 units.

Typical periods offered: Fall

CSE 4507 Introduction to Visualization

Disciplines such as medicine, business, science, and government are producing enormous amounts of data with increasing volume and complexity. In this course, students will study the principles for transforming abstract data into useful information visualizations. They will learn about the state of the art in visualization research and development and gain hands-on experience with designing and developing interactive visualization tools for the web. They will also learn how to critique existing visualizations and how to evaluate the systems they build. Topics include design, data mapping, visual perception, and interaction. Prerequisite: CSE 330S.

Credit 3 units.

Typical periods offered: Fall

CSE 4600 Switching Theory

Advanced topics in switching theory as employed in the synthesis, analysis, and design of information processing systems. Combinational techniques: minimization, multiple output networks, state identification and fault detection, hazards, testability and design for test are examined. Sequential techniques: synchronous circuits, machine minimization, optimal state assignment, asynchronous circuits, and built-in self-test techniques. Prerequisite: CSE 260M

Credit 3 units.

CSE 4602 Computer Systems Design

Introduction to modern design practices, including FPGA and PCB design methodologies. Student teams use Xilinx Vivado for HDL-based FPGA design and simulation and do schematic capture, PCB layout, fabrication, and testing of the hardware portion of a selected computation system. The software portion of the project uses Microsoft Visual Studio to develop a user interface and any additional support

software required to demonstrate final projects to the faculty during finals week. Prerequisites: CSE 361S and 362M from Washington University in St. Louis or permission of the instructor. Revised: 2019-02-22

Credit 3 units.

Typical periods offered: Spring

CSE 4603 Digital Integrated Circuit Design and Architecture

This is a project-oriented course on digital VLSI design. The course material focuses on bottom-up design of digital integrated circuits, starting from CMOS transistors, CMOS inverters, combinational circuits and sequential logic designs. Important design aspects of digital integrated circuits such as propagation delay, noise margins and power dissipation are covered in the class, and design challenges in sub-micron technology are addressed. The students design combinational and sequential circuits at various levels of abstraction using a state-of-the-art CAD environment provided by Cadence Design Systems. The goal of the course is to design a microprocessor in 0.5 micron technology that will be fabricated by a semiconductor foundry. Prerequisites: CSE 260M and ESE 232.

Credit 3 units.

Typical periods offered: Spring

CSE 4607 Embedded Computing Systems

This course introduces the issues, challenges, and methods for designing embedded computing systems -- systems designed to serve a particular application and which incorporate the use of digital processing devices. Examples of embedded systems include cellular phones, appliances, game consoles, automobiles, and drones. Emphasis is given to aspects of design that are distinct to embedded systems. The course examines hardware, software, and system-level design. Hardware topics include microcontrollers, digital signal processors, memory hierarchy, and I/O. Software issues include languages, run-time environments, and program analysis. System-level topics include real-time operating systems, scheduling, power management, and wireless sensor networks. Students will perform a course project on a real embedded system testbed. Prerequisites: CSE 260M (and either CSE 132 or ESE 205).

Credit 3 units.

Typical periods offered: Fall

CSE 4608 Introduction to Quantum Computing

This course introduces students to quantum computing, which leverages the effects of quantum-mechanical phenomena to solve problems. The course begins with material from physics that demonstrates the presence of quantum effects. Mathematical abstractions of quantum gates are studied with the goal of developing the skill to reason about existing quantum circuits and to develop new quantum circuits as required to solve problems. Students will develop a quantum-computer simulator and make use of open simulators as well as actual devices that can realize quantum circuits on the Internet. Prerequisites: CSE247, Math 309 and Math 3200 (or ESE 326 or Engr 328).

Credit 3 units.

Typical periods offered: Fall, Spring

CSE 4690 Security of the Internet of Things and Embedded System Security

This course presents a deep dive into the emerging world of the Internet of Things from a cybersecurity perspective. With billions of internet-enabled devices projected to impact every nook and cranny of modern existence, the concomitant security challenge portends to become dazzlingly complex. Students will learn the fundamentals of IoT architecture and operations from a layered perspective and focus on identifying, assessing, and mitigating the threats and vulnerabilities

therein. Through a blend of lecture and hands-on studios, students will gain proficiency in the range of approaches, methods, and techniques required to address embedded systems security and secure the internet of things using actual devices from both hardware and software perspectives and across a range of applications.
Credit 3 units.

CSE 4703 Introduction to Computer Networks

A broad overview of computer networking. Topics include history, protocols, Hyper Text Transfer Protocol (HTTP), File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), Domain Name System (DNS), peer-to-peer (P2P), transport layer design issues, transport layer protocols, Transmission Control Protocol (TCP), User Datagram Protocol (UDP), TCP congestion control, network layer, Internet Protocol version 4 (IPv4), Internet Control Message Protocol (ICMP), Internet Protocol version 6 (IPv6), routing algorithms, routing protocols, Open Shortest Path First (OSPF), Routing Information Protocol (RIP), Border Gateway Protocol (BGP), datalink layer and local area networks carrier sense multiple access with collision detection (CSMA/CD), Ethernet, virtual local area networks (VLANs), Point-to-Point Protocol (PPP), Multi-Protocol Label Switching, wireless and mobile networks, multimedia networking, security in computer networks, cryptography, and network management. Prerequisite: CSE 247; CSE 132 is suggested, but not required. Revised: 2019-02-22
Credit 3 units.
Typical periods offered: Fall, Spring

CSE 4970 Senior Project I

Implementation of a substantive project on an individual basis, involving one or more major areas in computer science. Problems pursued under this framework may be predominantly analytical, involving exploration and extension of theoretical structures, or may pivot around the design/development of solutions for particular applications drawn from areas throughout the University and/or community. In either case, the project serves as a focal point for crystallizing the concepts, techniques, and methodologies encountered throughout the curriculum. Students intending to take CSE 497-498 must submit a project proposal form (<https://cse.wustl.edu/undergraduate/PublishingImages/Pages/undergraduate-research/Independent%20Study%20Form%20400.pdf>) for approval by the department during the spring semester of the junior year. Prerequisite: senior standing.
Credit 3 units.
Typical periods offered: Fall, Spring

CSE 4971 Senior Project II

Implementation of a substantive project on an individual basis, involving one or more major areas in computer science. Problems pursued under this framework may be predominantly analytical, involving exploration and extension of theoretical structures, or may pivot around the design/development of solutions for particular applications drawn from areas throughout the University and/or community. In either case, the project serves as a focal point for crystallizing the concepts, techniques, and methodologies encountered throughout the curriculum. Students intending to take CSE 497-498 must submit a project proposal form (<https://cse.wustl.edu/undergraduate/PublishingImages/Pages/undergraduate-research/Independent%20Study%20Form%20400.pdf>) for approval by the department during the spring semester of the junior year. Prerequisite: senior standing.
Credit 3 units.
Typical periods offered: Fall, Spring

CSE 4975 Undergraduate Honors Thesis

Working closely with a faculty member, the student investigates an original idea (algorithm, model technique, etc.), including a study of its possible implications, its potential application, and its relationship to previous related work reported in the literature. Contributions and results from this investigation are synthesized and compiled into a publication-quality research paper presenting the new idea. Prerequisites: a strong academic record and permission of instructor.
Credit 3 units.
Typical periods offered: Fall, Spring
