

# COMPUTER SCIENCE AND ENGINEERING

The mission of Fairfield University is to educate its students through a variety of scholarly and professional disciplines. It offers opportunities for individual and common reflection, and it provides training in such essential human skills as analysis, synthesis, and communication. Computational thinking and processes permeate our daily lives, revolutionizing our understanding of the natural world, our tools, and of ourselves. Knowledge of computer science has become highly valued in such diverse fields as psychology, biology, and even philosophy. A degree in Computer Science gives one both marketable skills and intellectual breadth that can be applied to many career choices. At Fairfield, students can pursue multiple degree options as follows: a BA degree in Computer Science, a BS degree in Computer Science, or a minor in Computer Science that makes a strong addition to one's resume. Students pursuing a BA degree in Computer Science can readily integrate a math major into their program by contacting the Mathematics department.

The BS in Computer Science is accredited by the Computing Accreditation Commission of the Accreditation Board for Engineering and Technology (ABET), <http://www.abet.org>. Additionally, the BS in Computer Science can be continued with a specialization in Software Engineering through a five-year Bachelor's and Master's accelerated degree track. The BA in Computer Science can be continued with a specialization in a Data Science Master's degree program.

Fairfield recognizes that learning is a life-long process and sees the education which it provides as the foundation upon which its students may continue to build within their chosen areas of scholarly study or professional development.

The Program Educational Objectives are broad statements that describe what alumni do within a few years following graduation. The BS in Computer Science major is committed to graduating computer scientists who within a few years of their graduation are expected to:

1. Apply their in-depth understanding in areas of computer science and engineering to solve problems in a modern technological society as productive computer scientists or engineers and/or enter and succeed in a graduate program.
2. Function effectively, both individually and within multi-disciplinary teams.
3. Continue as a lifelong learner to develop their computing and engineering abilities, problem-solving skills, and aptitude for innovation.
4. Practice professional ethics with social and cyber responsibility through service in the framework of a global technical community.

The Computer Science students pursuing a BS degree will graduate from the program with the ability to perform and execute the following student outcomes:

1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
3. Communicate effectively in a variety of professional contexts.

4. Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.
5. Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.
6. Apply computer science theory and software development fundamentals to produce computing-based solutions.

## Programs

- Computer Science Major (BA)
  - Concentration in Computer Engineering
  - Concentration in Software Engineering
- Computer Science Major (BS)
  - Concentration in Computer Engineering
  - Concentration in Software Engineering
- Computer Science Minor
- Dual Major in Computer Science and Finance

- 
- Data Science Five-Year Accelerated Degree Bachelor and Master of Science Program
  - Software Engineering Five-Year Accelerated Degree Bachelor and Master of Science Program

## Courses

### Computer Science

#### CPSC 1101 Introduction to Computing

3 Credits

In this course, students learn computational problem-solving techniques through the process of design, implementation, testing, and documentation using the programming language Python. The main ideas of computing are explored and students learn the most essential information about computers and technology in today's digital world and the latest computing trends and skills. Students will get an understanding of the breadth of computing as a discipline and how it exists in the world by identifying computing applications in society and exposing them to a variety of computing topics. Previously CS 0101.

#### CPSC 1131 Fundamentals of Programming

3 Credits

**Attributes:** BUEL Business Elective, ENPC Digital Journalism Production Component

This course introduces programming constructs and techniques in a logical progression beginning with small problems and basic algorithms through larger scale programs and design. While not an object oriented course, classes and objects are used in an ancillary capacity while working on broader topics of software architecture. Complete programs will be designed, coded, and debugged in both Java and the C programming language, developing skills necessary to work with more complex software systems. Previously CS 0131.

#### CPSC 1152 Introduction to Computer Game Modeling

3 Credits

This is an introductory computer games modeling course which examines the basics of computer game design and visual effects. Students will use graphics software modeling packages to create characters and visual effects, and to develop a computer game idea, including storyline and plots. Basic programming techniques may also be taught. Previously CS 0152.

**CPSC 1201 Technical Skills for Liberal Arts Majors 3 Credits**

Students today will need skills in a variety of computer programs and software applications. This course is designed for those in the liberal arts (humanities and social or behavioral sciences) who will need these technical skills for their future workplace. Specific skill sets include software for word processing, spreadsheet, presentation, conferencing, web page coding, and web page design; other software may be included. The course serves those students going into business, publishing, non-profit careers, and a variety of other enterprises for which the liberal arts training must be augmented with specific cyber-related technical competence. Open to students with majors in the humanities or social/behavioral sciences only. Previously CS 0201.

**CPSC 2231 Programming Workshop 3 Credits**

**Corequisite:** CPSC 2231L.

**Prerequisite:** CPSC 1131.

This course covers advanced programming concepts in one or more current programming languages, including syntax and theories. It prepares students for adapting to various programming environments and coding in an efficient manner. Lab work will accompany the course. Previously CS 0231.

**CPSC 2231L Programming Workshop Lab 1 Credit**

**Corequisite:** CPSC 2231.

This lab accompanies the Programming Workshop course for hands-on practice with course concepts. Previously CS 0231L.

**CPSC 2232 Data Structures 3 Credits**

**Corequisite:** CPSC 2232L.

**Prerequisite:** CPSC 2231.

This course presents problem solving with abstract data types such as linked lists, stacks, queues, trees, heaps, and graphs. The course revisits recursion and discusses algorithm efficiency. The course may also include sorting, reachability, and minimal paths in graphs and their algorithms. Students apply data structure concepts in advanced programming. Previously CS 0232.

**CPSC 2232L Data Structures Lab 1 Credit**

**Corequisite:** CPSC 2232.

This lab accompanies the Data Structures course for hands-on practice with course concepts. Previously CS 0232L.

**CPSC 2250L Computer Science Sophomore Clinic 1 Credit**

**Corequisite:** CPSC 2304.

This sophomore clinic provides faculty guidance and supervision beyond the scope of existing courses. Students learn how to develop and structure their deliverables, as well as how to use computer science tools in the context of real-world or research projects. Previously CS 0250L.

**CPSC 2304 Web Development 3 Credits**

**Attributes:** ENPC Digital Journalism Production Component

**Prerequisite:** CPSC 1131.

This course introduces the student to developing applications for use on the World Wide Web. Students learn basic n-tier concepts for designing distributed applications and gain hands on experience through the construction of web-based applications. The course covers concepts that allow communication over the Web. This includes designing and authoring web pages, markup languages, the client-side document object model, usability, search engine optimization, and client-side dynamic web pages. Graduate equivalent: SWEG 5304. Previously CS 0304.

**CPSC 3333 Introduction to Cybersecurity 3 Credits**

**Prerequisite:** CPSC 2231.

In this course, students will be given an extensive overview of the various components of cybersecurity, including software development, operating systems, databases, and networks. Students will learn cybersecurity concepts, issues, and tools that are critical in solving problems in the computing security domain. The course will use lectures, reading assignments, and interactive lab exercises to reinforce the concepts that are introduced. Graduate equivalent: SWEG 5333. Previously CS 0333.

**CPSC 3343 Design and Analysis of Algorithms 3 Credits**

**Prerequisite:** CPSC 2232.

This course focuses on algorithmic problem-solving techniques including decrease-and-conquer, divide-and-conquer, transform-and-conquer, dynamic programming, greedy design, backtracking, branch and bound, and heuristic algorithms. Advanced data structure techniques are studied within the context of designing efficient graph algorithms. Students gain an appreciation for the fundamental notion of NP completeness. The asymptotic analysis of algorithmic complexity is emphasized throughout the course. Previously CS 0343.

**CPSC 3349 Cloud Computing 3 Credits**

**Prerequisite:** Junior standing.

This course will introduce the foundations of cloud computing, and familiarize students with the core concepts needed to build, deploy and manage applications in a cloud. Besides the theoretical underpinnings, emphasis will be put on practical experience of using cloud resources and services. Concepts like microservices and containers will be discussed in depth, as well as best practices for building successful cloud native applications and implications for development and operational processes. The course will be a combination of lectures and hands-on experience of a public cloud. Graduate equivalent: SWEG 5349.

**CPSC 3351L Computer Science Junior Clinic I 1 Credit**

**Corequisite:** SWEG 3301.

This first junior clinic provides faculty guidance and supervision beyond the scope of existing courses. Students learn how to develop and structure their deliverables, as well as how to use computer science tools in the context of real-world or research projects. Previously CS 0350L.

**CPSC 3352L Computer Science Junior Clinic II 1 Credit**

**Corequisite:** SWEG 3302.

This second junior clinic provides faculty guidance and supervision beyond the scope of existing courses. Students learn how to develop and structure their deliverables, as well as how to use computer science tools in the context of real-world or research projects. Previously CS 0351L.

**CPSC 3354 Theory of Programming Languages 3 Credits**

**Prerequisite:** CPSC 2232.

The objective of this course is to provide a better understanding of programming languages and their design. Various concepts and principles underlying the design and use of modern programming languages are considered, mainly: syntax, semantics, type systems, runtime environment. Students take a pragmatic approach to this course which ties in the theory and real-world implications in some selected programming languages. The course has a substantial theoretical component which is supported by programming assignments and projects. Previously CS 0354.

- CPSC 4305 Mobile Application Development** 3 Credits  
**Prerequisite:** CPSC 1131.  
 This project-oriented course examines the fundamental aspects of mobile computing, application architecture, and mobile application design and development. Students will learn application development on the Android platform. Students will complete a hands-on project building a prototype mobile application. Topics include user interface design and building, input and data handling, and network techniques and GPS and motion sensing. Students are expected to work on a project that produces a professional-quality mobile application. Projects will be deployed in real-world applications. Graduate equivalent: SWEG 5305. Previously CS 0305.
- CPSC 4314 Network Security** 3 Credits  
 This course is intended for individuals who need an understanding of the client-server environment, with any emphasis on network security. The OSI Model, network concepts, and network architecture are discussed. The components that make up a network, including cabling, wiring hubs, file servers, bridges, switches, routers, network interface cards, network operating systems, and network software and hardware configurations are discussed. Network architectural concepts, wide area networks, remote access, and segmentation are discussed. Operating systems will be discussed and demonstrated. Featured is the seven-layer OSI model, the foundation of today's communication protocols. Students will work with various security protocols and configure routers and switches with security methods. Previously CS 0314.
- CPSC 4315 Computational Biology** 3 Credits  
**Prerequisite:** Junior standing.  
 This course is designed to benefit computational and experimental biologists to understand the principles of analyzing biological data, building models and testing hypotheses using computer science paradigms. Students will learn how to build computational tools that are used to analyze DNA content, identify protein binding patterns, compare sequences, and discover variation within genomes. Graduate equivalent: SWEG 5315. Previously CS 0315.
- CPSC 4317 Computational Statistics for Biomedical Sciences** 3 Credits  
**Prerequisite:** Junior standing.  
 This course will provide a practical introduction to analysis of biological and biomedical data. Basic statistical and machine learning techniques will be covered, including descriptive statistics, linear regression, non-linear regression, classification/prediction, and biomedical data visualization. Emphasis will be on how to choose appropriate data analysis models and how to assess statistical significance. This course will benefit data scientists to apply data science techniques to analyze biomedical data or clinical data. In addition, this course is also designed to benefit computational and experimental biologists to understand the principles of analyzing biological data, building models and testing hypotheses using computer science paradigms. To visualize data and carry out data analysis, students will learn R or Python, and other programming languages for statistical computing and graphics. The class will be a combination of lecture and computer lab. Graduate equivalent: SWEG 5317. Previously CS 0317.
- CPSC 4322 Visual Analytics** 3 Credits  
**Prerequisite:** Junior standing.  
 In this course, students investigate visual analytics tools and techniques used to synthesize information and derive insight from massive, dynamic, ambiguous, and often conflicting data and to communicate the findings effectively for decision-making. Extensive use of case studies based on real-world events will be used to illustrate course concepts. Students will apply visual analytics techniques toward a focused research problem in a real-world application or a domain of interest. Graduate equivalent: SWEG 5322. Previously CS 0320.
- CPSC 4331 Operating Systems** 3 Credits  
**Prerequisite:** CPSC 2232.  
 This course introduces students to the basic operating system structure, process and thread synchronization and concurrency, file systems and storage servers, memory management techniques, process scheduling and resource management, virtualization, and more on a UNIX-like platform. Previously CS 0331.
- CPSC 4335 Digital Forensics** 3 Credits  
**Prerequisite:** Junior standing.  
 In this course students will be given the basic notions and theory of digital forensics. For file systems and operating systems, the class covers investigative techniques and legal and technical considerations that the examiner should make. They will learn concepts, challenges, and tools in applying digital forensics examinations. The course includes, but not limited to, topics in the suggested curriculum of CDFE certification. The course will use lectures, reading assignments, and interactive lab exercises to reinforce the concepts that are introduced. Graduate equivalent: SWEG 5335.
- CPSC 4350 Introduction to Data Science** 3 Credits  
**Prerequisite:** CPSC 1101, Junior or Senior Standing or instructor permission.  
 This course offers a thorough introduction to data science, focusing on both practical skills and theoretical knowledge across a wide range of topics. It guides students through the entire data science lifecycle, starting from the basics of data wrangling, exploratory data analysis, and visualization, to more advanced topics such as statistical inference, machine learning, and natural language processing. Python with web-based interactive computing platforms, such as JupyterLab, are utilized as the primary tools for hands-on learning. Through a combination of lectures, hands-on projects, and assessments, students will learn to derive insights from data and make informed decisions based on data analysis. Graduate equivalent: SWEG 5350.
- CPSC 4355 Artificial Intelligence** 3 Credits  
**Prerequisite:** CPSC 2232.  
 This course provides an overview of methods, history, and typical applications of AI. It covers problem solving, machine learning, probabilistic reasoning, classical search algorithms, deep learning, applications (natural language processing, vision). Students solve a variety of AI problems using Python. Students are able to understand the nature of the intelligence behaviors, develop an AI system, grasp the classical approaches to AI research, and apply these approaches and tools for real problem solving. Graduate equivalent: SWEG 5355. Previously CS 0355.
- CPSC 4357 Database Management Systems** 3 Credits  
**Prerequisite:** CPSC 2232.  
 This course examines data management systems; relational database model; domains and relational integrity; structured query language (SQL); database design, logical and physical; entity-relationship diagrams; normalization; transaction processing; and database administration. Students perform a number of hands-on exercises using an industry-standard database for modern applications. Graduate equivalent: SWEG 5357. Previously CS 0357.

**CPSC 4360 Machine Learning****3 Credits****Prerequisite:** Junior standing.

This course will provide a practical introduction to machine learning applications such as face recognition, clinical diagnosis, speech recognition, natural language processing, or image classification. Topics such as regression, classification, neural networks, deep learning, and ensemble methods will be discussed. Emphasis will be on how to choose appropriate machine learning and deep learning models and how to evaluate their performance. The class will be a combination of lecture and computer lab. Graduate equivalent: SWEG 5360.

**CPSC 4366 Deep Learning****3 Credits****Prerequisite:** CPSC 1101, CPSC 4360, Junior or Senior Standing.

This course offers a comprehensive introduction to the fundamental principles, theories, and practical aspects of deep learning. Lectures will begin with the basics of shallow neural networks before progressing to complex deep neural network structures. The topics include an overview of key neural network architectures, such as convolutional, recurrent, and autoencoder networks, and their applications in computer vision for tasks like image classification and segmentation, as well as in natural language processing for text classification and machine translation. Additionally, students will learn to craft deep learning architectures using Python on open-source machine learning platforms. Through a combination of lectures, practical exercises, and final projects, students will acquire the ability to implement deep learning models in real-world situations. Graduate equivalent: SWEG 5366.

**CPSC 4521 Information Visualization****3 Credits****Prerequisite:** Junior standing.

This course introduces basic elements of Information Visualization, which is concerned with the creation of visual representations of Big Data abstract phenomena for which there may not be a natural physical reality, such as stock market movements, social relationships, gene expression levels, manufacturing production monitoring, survey data from political polls, or supermarket purchases. Students will be exposed to techniques covering the five main phases of developing information visualization tools: representation, presentation, interaction, perception and interpretation, and evaluation. Students will be required to develop a large project related to information visualization. Graduate equivalent: SWEG 5521.

**CPSC 4525 Human Computer Interaction****3 Credits****Prerequisite:** Junior standing.

This course introduces students to the foundations of Human Computer Interaction and how it applies in software engineering and research settings. Students will learn how to design user interfaces based on the capabilities of computer technology and the needs of human factors. They will design user interfaces and learn how to implement a prototype from a list of informal requirements. It will also introduce students to issues related to human subject research as well as ethical implications of human computer interaction. Graduate equivalent: SWEG 5525.

## Software Engineering

**SWEG 3301 Software Engineering Methods****3 Credits****Attributes:** MWID Magis Core: Writing in the Discipline**Corequisite:** CPSC 3351L.**Prerequisite:** CPSC 2232.

This course explores the requirements gathering, system analysis, and software design methods of software application following the software processes required for the production of high quality software. Techniques for creating documentation and using software development tools will be presented. Students will gain experience in software project management, requirements, analysis, and safety issues in software development, interpersonal skills for management and team membership, and the software engineering discernment of systems architecture. Graduate equivalent: SWEG 5301. Previously SW 0300.

**SWEG 3302 Software Design Methods****3 Credits****Prerequisite:** SWEG 3301.

This course is the continuation of SWEG 3301 with in-depth projects and further discussions of design and implementation topics. Through the use of case studies and project work that has the student gradually building a large design specification, students will achieve an understanding of how complex applications are designed and built. Graduate equivalent: SWEG 5302. Previously SW 0301.

**SWEG 4312 Agile Software Engineering****3 Credits****Prerequisite:** SWEG 3301.

In this course, students apply in-depth techniques and experience various roles incorporated into one of the main approaches to software development which is agile methodology. It uses detailed knowledge about each of the major traditional software engineering phases to explore a more iterative approach for development of faster and more adaptable software. Proficiency in programming is expected of the students entering this course. Graduate equivalent: SWEG 5312. Previously SW 0312.

**SWEG 4320 Software Testing and Maintenance****3 Credits****Prerequisite:** SWEG 3301.

This course will cover in-depth methods for software testing, reliability and maintenance of software. Students will learn the principles of software testing and how to apply software testing techniques to the development of quality software and how to deploy software systems, maintain, enhance, and reuse software systems. Graduate equivalent: SWEG 5320. Previously SW 0320.

**SWEG 4321 Software Project Management****3 Credits****Prerequisite:** SWEG 3301.

This course explores and practices fundamental project management skills and life cycles required for both the successful management and development of software. Quality management principles of Personal Software Process (PSP) and Team Software Process (TSP) are introduced and practiced. Students will learn how to develop a project plan, scope a project, identify project activities, create work breakdown structures, estimate and schedule resources, construct and analyze project network diagrams, finalize project schedule and cost based on resource activity, recruit team members, organize and manage a project team, monitor and control progress, understand critical path project management, and have knowledge of both agile and traditional project management methods. Graduate equivalent: SWEG 5321. Previously SW 0321.

**SWEG 4505 Advanced Database Concepts 3 Credits**

This course covers topics in database implementation designed to provide software engineers with a wide variety of server-side problem solving techniques. Topics include cursors, query and index optimization, advanced SQL programming, distributed databases, object-oriented databases, clustering, partitioning, and working with XML and other unstructured data. While Microsoft SQL Server is primarily used for demonstration, the topics covered are applicable to any database platform, and the different approaches of the major database vendors are frequently contrasted. The format consists of lecture and lab components. Cross-listed with SWEG 6505.

**SWEG 4599 Ethical Hacking 3 Credits**

This course covers current information security practices and countermeasures put in place to safeguard against security breaches. The course reviews internet infrastructures such as firewalls, IDS systems, and honey pots. Additional areas include risk analysis, computer-use policies, physical security, internet/intranet security, malware, firewall infrastructure, and current information security issues. Cross-listed with SWEG 6599.

**SWEG 4900 Special Topics (Shell) 3 Credits**

This course provides an in-depth study of selected topics in software engineering of particular interest to the students and instructor. The course is counted as a major elective/specialization course. The topics and prerequisites will be announced when this course is offered. Previously SW 0382.

**SWEG 4990 Independent Study 1-3 Credits**

This course is an individualized study under the supervision of the faculty member. The course emphasizes individual creativity. Undergraduate students work with a faculty mentor in studying and investigating topics of current interest in software engineering. Enrollment by permission only. Previously SW 0383.

## Faculty

### Professors

Yang, chair

### Associate Professors

Rusu

### Assistant Professors

Bandara  
Mathur

### Assistant Professors of the Practice

Majumder  
Speretta

### Instructor of the Practice

Kramer

### Lecturers

Corcoran  
Galasso  
Wilson