

# Computer Science

## Information Access Policy

### Clemson University Libraries

Written by Jan Comfort ([comforj@clemson.edu](mailto:comforj@clemson.edu)) 4/2017  
based on policy written by J. Groff, ([jgroff@clemson.edu](mailto:jgroff@clemson.edu)) Spring 2012

#### Introduction

This Information Access Policy is a statement of goals for building the library's collection in the subject area of Computer Science. It should serve as a guide for library personnel in making collection development decisions, and should inform users, in general, how library materials are selected.

#### Purpose of Collection

The Computer Science collection supports the research and teaching needs of undergraduate students, graduate students, faculty, and staff in the School of Computing at Clemson. Additionally, the collection supports collaborative research with other departments in the College of Engineering, Computing and Applied Sciences (CECAS) and with other colleges at Clemson. The computing needs of the University as a whole are also considered.

#### Clemson's School of Computing

Clemson's School of Computing was formed from the Department of Computer Science in 2007 to integrate computation with the arts, sciences, and engineering. The School has three primary academic divisions: computer science, visual computing, and human-centered computing.

Undergraduate students may earn a B.S. or B.A. degree in Computer Science, or a B.S. in Computer Information Systems. Students can minor in Computer Science and Digital Production Arts. Graduate degrees include a Ph.D. in Computer Science, Ph.D. in Human-Centered Computing, M.S. in Computer Science, and an M.F.A. in Digital Production Arts. The School of Computing also offers a Systems Engineering Graduate Certificate Program for working professionals.

The Computer Science curriculum is ABET-accredited. It's important to note that the *Facilities* section of ABET's *Criteria for Accrediting Computing Programs, 2012-2013*, states that library services and the computing and information infrastructure must be adequate to support the scholarly and professional activities of the students and faculty.

#### Collection Locations

The majority of the physical collection is in Cooper Library. Some older books and print journal volumes, as well as more recent print journal volumes duplicated in a stable online format (such as Science Direct titles), are available in Offsite Shelving.

## General Collection Guidelines

**Languages:** English is the primary language of the collection. Materials in languages other than English are not actively selected.

**Geographic Guidelines:** No geographical limitations will be placed on the acquisition of Computer Science materials.

**Chronological Guidelines:** Emphasis is on current research and development, but retrospective collecting may also be done at the request of faculty in support of their research and teaching.

**Types of Materials Selected:** Emphasis is on research materials and materials to support the curricula of Clemson's School of Computing, and the curricula in related disciplines. The collection includes monographs, journals and other serials, reference works, indexes and abstracts, and data collections.

Graduate level texts and books written on specific programming languages are acquired extensively, while basic undergraduate textbooks whose primary function is instruction are usually excluded.

Biographies and works on the history of Computer Science are acquired very selectively. Materials published by Clemson University authors will be purchased as requested. Books on popular software packages and programming languages will be purchased selectively. Industry standards and/or technical reports will be purchased for individuals as requested.

Indexes and abstracts are licensed as online databases; the ACM Digital Library, IEEE Xplore, Engineering Village (Includes INSPEC, Compendex, & Engineering Index backfiles), and Web of Science are several examples of subscribed resources most used by researchers in Computer Science. Given that access to computer science journals is key to the success of Clemson researchers across colleges and departments at Clemson, every effort will be made to increase the size of Clemson's journals collection in Computer Science.

**Format of Materials:** No format is excluded. For journals, indexes, reference works, and data collections, the online versions are preferred. Books and book series are increasingly purchased in electronic format. eBook packages from major vendors are particularly important. Duplication of formats will largely be avoided.

**Library of Congress Classifications:** The predominant LC classifications of the collection are QA75-76, but supporting materials may also be purchased in other LC Subclasses.

Computer Science	
LC Class	Subject
QA75-76.95	Calculating machines
QA75.5-76.95	Electronic computers. Computer science
QA76.75-76.765	Computer software

## Selection Tools

Faculty input and research interests will be given the highest priority in selection decisions. In addition, circulation statistics, publisher catalogs, and book reviews will be used in selecting monographs. Impact factors and available journal usage statistics may play a role in journal selection decisions. Interlibrary Loan requests will also be monitored so that materials frequently requested from other institutions may be purchased.

### Selection, Evaluation, and Assessment Tools

- Faculty requests and recommendations
- Faculty requests forwarded from ILL as books currently in print and appropriate for the collection
- Approval notifications from YBP identified as titles that fit the needs of the library's collection
- Vendor databases (such as YBP GOBI, and ScienceDirect)
- Review sources, e.g. *Choice*, *Books for College Libraries*, and *Magazines for Libraries*
- Indexes, bibliographies, and series lists
- Review of the teaching and research interests of the Electrical and Computer Engineering faculty
- Interlibrary loan and circulation activity
- Publishers' catalogs
- Usage statistics for databases, e-journals and e-books

## Access to Materials not Available at Clemson

Interlibrary Loan is the primary method of accessing both monographic materials and published articles. Books available from other research institutions in South Carolina can be supplied by [Pascal Delivers](#), also free to Clemson University students, faculty and staff.

## LibGuides

A Library Guide for Computer Science is linked on the Library's Research and Course Guides page <http://clemson.libguides.com/> as well as the Research page <http://libraries.clemson.edu/research/>

## Weeding

Monographic material published ten years prior to the current date that has not circulated within the last five years will be considered for weeding. Older materials that have not circulated recently may be kept for historical and/or reference purposes.

Books will be removed from the collection if they are in poor condition and cannot be repaired. Every effort will be made to replace high-use titles.

Print serials may be sent to Offsite Shelving if they are duplicated in a stable online format.

## Appendix One – Undergraduate Program

### Bachelors Degree Programs

The School of Computing offers three undergraduate degree programs each containing a strong and common technical core that emphasizes the development of programming and problem solving skills:

- **B.S. in Computer Science**, accredited by the Computing Accreditation Commission of [ABET, Inc.](#)  
[\[Program Requirements\]](#) [\[Sample Four Year Schedule\]](#)
- **B.A. in Computer Science**  
[\[Program Requirements\]](#) [\[Sample Four Year Schedule\]](#)
- **B.S. in Computer Information Systems**  
[\[Program Requirements\]](#) [\[Sample Four Year Schedule\]](#) [\[Additional College of Business Fees\]](#)
- **Side-by-Side Comparison** [\[Program Comparison\]](#)

The programs share a common computer science core, but typically differ in requirements during the junior and senior years with different emphases in areas outside of computer science: The BS-CS degree requires more required mathematics, science, and computer science coursework.

- The BA-CS degree requires four semesters of a foreign language course and a minor.
- The BS-CIS degree requires business, accounting, and information systems coursework.

### ***Combined Bachelors/Masters Degree Program***

The School of Computing offers a combined BS-MS education plan which allows students to apply up to nine hour of graduate (6000- and 8000-level courses) towards both the BS and MS degrees. Participating students must have at least 90 semester hours (junior standing) and a minimum GPA of 3.4.

### ***Honors Program***

Departmental Honors in computer science requires taking a junior-year honors seminar and a senior-year honors thesis (for a total of 7-8 credit hours). Please see Mrs. Gail Grieger or Dr. Chris Plaue for honors advising. The previous honors advisor, Dr. Jim Martin, also maintains [additional information](#) and frequently asked questions.

### ***Minors***

The School of Computing offers two minor programs to allow Clemson students the opportunity to add a technical component to their degree:

- **Computer Science**
- **Digital Production Arts**

## Undergraduate Courses in Computer Science

**CPSC 1010/1011, Computer Science I 4 (3,2)** Introduction to modern problem solving and programming methods. Special emphasis is placed on algorithm development and software life cycle concepts. Includes use of appropriate tools and discusses ethical issues arising from the impact of computing upon society. Intended for students concentrating in computer science or related fields. Includes Honors sections. Preq or concurrent enrollment: MATH 1020 or MATH 1040 or MATH 1050 or MATH 1060 or MATH 1080 or MATH 2070. Students who do not meet the prerequisite, but who score a satisfactory score on the Clemson Mathematics Placement Test, or have AP or transfer credit for their math requirements, may request a registration override from the instructor.

**CPSC 1020/1021, Computer Science II 4 (3,2)** Continuation of CPSC 1010. Continued emphasis on problem solving and program development techniques. Examines typical numerical, nonnumerical, and data processing problems. Introduces basic data structures.

**CPSC 1040/1041 Introduction to the Concepts and Logic of Computer Programming 2 (1,2)** Intro to the concepts and logic of computer programming. Simple models are used to introduce basic techniques for developing a programmed solution to a given problem. Problem solving techniques are considered.

**CPSC 1110/1111 Introduction to Programming in C 3 (2,2)** Introduction to computer programming in C and its use in solving problems. Intended primarily for technical majors. Basic instruction in programming techniques, algorithms and standard Unix software development tools and utilities

**CPSC 1150 Introduction to Computational Science 3 (3,0)** Intro to systems thinking. Includes development of dynamical systems models using visual modeling tools and development of dynamical systems using agent based software. Class material investigates elementary science and engineering models.

**CPSC 1200/1201 Introduction to Information Technology 3 (2,2)** Investigation of ethical and societal issues based on the expanding integration of computers into our everyday lives. Considers historical background, terminology, new technologies and the projected future of computers. Includes practical experience with common computer software technologies.

**CPSC 1610 Introduction to Visual Basic Programming 3 (2,2)** Introduction to programming using the Visual Basic language. Topics include simple and complex data types, arithmetic operations, control flow, files, and database programming. Several projects are implemented during the semester.

**CPSC 1990 Creative Inquiry in Computing 1-3 (1-3)** Students engage in faculty-led research in the context of a team effort. May be repeated for a maximum of six credits.

**CPSC 2070 Discrete Structures for Computing 3 (3)** Introduces ideas and techniques from discrete structures that are widely used in the computing sciences. Topics emphasize techniques of rigorous argumentation and application to the computing disciplines.

**CPSC 2100/2101 Programming Methodology 4 (3,2)** Introduction to programming techniques and methodology. Topics include structured programming, stepwise refinement, program design and implementation techniques, modularization criteria, program testing and verification, basic data structures, and analysis of algorithms. Credit may not be received for both CPSC 1020 and 2100.

**CPSC 2120/2121 Algorithms and Data Structures 4 (3,2)** Study of data structures and algorithms fundamental to computer science; abstract data-type concepts; measures of program running time and time complexity; algorithm analysis and design techniques.

**CPSC 2150/2151 Software Development Foundations 3 (2,2)** Intensive study of software development foundations. Advanced coverage of programming language primitives, function-level design principles, and standard development and debugging tools. Introductory coverage of module-level design principles, program specification and reasoning principles, and validation and verification techniques.

**CPSC 2200 Microcomputer Applications 3 (3)** Applications of microcomputers to formulate and solve problem models. Emphasizes applications development in database and spreadsheet environments. Current software products are used. Students are expected to have experience with word processing and spreadsheet applications.

**CPSC 2310/2311 Introduction to Computer Organization 4 (3,2)** Study of the machine architectures on which algorithms are implemented and requirements of architectures that support high-level languages, programming environments, and applications.

**CPSC 2810 Selected Topics in Computer Science 1-4 (1-4)** Areas of computer science in which new trends arise. Innovative approaches to a variety of problems in the use and understanding of basic computing concepts are developed and implemented.

**CPSC 2910 Seminar in Professional Issues I 1 (1)** Considers the impact of computer use on society. Discusses ethical use of software and protection of intellectual property rights. Profession is viewed historically; organizations important to the profession are discussed; the development process for standards is presented; and students are introduced to the professional literature.

**CPSC (ECE) 3220 Introduction to Operating Systems 3 (3)** Detailed study of management techniques for the control of computer hardware resources. Topics include interrupt systems, primitive level characteristics of hardware and the management of memory, processor, devices, and data.

**CPSC 3300 Computer Systems Organization 3 (3)** Introduction to the structure of computer systems. Various hardware/software configurations are explored and presented as integrated systems. Topics include digital logic, basic computer organization, computer arithmetic, memory organization, input/output organizations, interrupt processing, multiprocessors, and cluster computers.

**CPSC 3500 Foundations of Computer Science 3 (3)** Development of the theoretical foundations of programming, algorithms, languages, automata, computability, complexity, data structures, and operating systems; a broad range of fundamental topics is consolidated and extended in prep for further study.

**CPSC (ECE) 3520 Programming Systems 3 (3)** Second course in programming languages and systems. Topics include assemblers, compilers, and syntactical methods; string manipulation and list processing; concepts of executive programs and operating systems; introduction to time-sharing systems.

**CPSC 3600 Networks and Network Programming 3 (3)** Introduction to basic concepts of computer network technologies and network programming. Topics include network programming, layered protocol architectures, local and wide area networks, internetwork and intranetwork concepts, security. Socket level programming is introduced and used throughout the course.

**CPSC 3620 Distributed and Cluster Computing 3 (3)** Introduction to the basic technology of and programming techniques for distributed and cluster computing. Standard techniques for developing parallel solutions to problems are introduced and implemented. Software systems that provide highlevel abstractions for data communications are considered.

**CPSC 3710 Systems Analysis 3 (3)** Incorporates a study of the decision-making process at all levels with the logical design of information systems. Extensive study of the system life cycle with emphasis on current as well as classical techniques for describing data flows, data structures, file design, etc.

**CPSC 3720 Introduction to Software Engineering 3 (3)** Intensive introduction to software engineering. Focuses on each major phase of the software lifecycle. Introductory coverage of requirements analysis, requirements modeling, design modeling, and project management. Intermediate coverage of module-level design principles, program specification and reasoning principles, and program validation and verification techniques.

**CPSC 3950 Honors Seminar 1 (1)** Research topics in various areas of computer science are presented. Methods for identifying and initiating research projects are considered.

**CPSC 3990 Advanced Creative Inquiry in Computing 1-3 (1-3)** Upper-division students engage in faculty-led research in the context of a team effort. May be repeated for a maximum of six credits.

**CPSC 4040 Computer Graphics Images 3 (3)** Presents the theory and practice behind the generation and manipulation of two-dimensional digital images within a computer graphics context. Image representation and storage, sampling and reconstruction, color systems, affine and general warps, enhancement and morphology, compositing, morphing, and non-photorealistic transformations.

**CPSC 4050\* Computer Graphics 3 (3)** Computational, mathematical, physical and perceptual principles underlying the production of effective three-dimensional computer graphics imagery.

**CPSC 4110\* Virtual Reality Systems 3 (3)** Design and implementation of software systems necessary to create virtual environments. Discusses techniques for achieving real-time, dynamic display of photorealistic, synthetic images. Includes hands-on experience with electromagnetically-tracked, head-mounted displays and requires, as a final project, the design and construction of a virtual environment.

**CPSC 4120\* Eye Tracking Methodology and Applications 3 (3)** Intro to the human visual system; visual perception; eye movements; eye tracking systems and applications in psychology, industrial engineering, marketing, and computer science; hands-on experience with real time, corneal-reflection eye trackers, experimental issues. Final project requires the execution and analysis of an eye tracking experiment

**CPSC 4140\* Human and Computer Interaction 3 (3)** Survey of human and computer interaction, its literature, history, and techniques. Covers cognitive and social models and limitations, hardware and software interface components, design methods, support for design, and evaluation methods.

**CPSC 4160\* 2-D Game Engine Construction 3 (3)** Introduction to tools and techniques necessary to build 2-D games. Techniques draw from subject areas such as software engineering, algorithms, and artificial intelligence. Students employ techniques such as sprite animation, parallax scrolling, sound, AI incorporated into game sprites, and the construction of a game shell.

**CPSC 4200\* Computer Security Principles 3 (3)** Covers principles of information systems security, including security policies, cryptography, authentication, access control mechanisms, system evaluation models, auditing, and intrusion detection. Computer security system case studies are analyzed.

**CPSC 4240\* System Administration and Security 3 (3)** Covers topics related to the administration and security of computer systems. Primary emphasis is on the administration and security of contemporary operating systems.

**CPSC 4280\* Design and Implementation of Programming Languages 3 (3)** Overview of programming language structures and features and their implementation. Control and data structures found in various languages are studied. Also includes runtime organization and environment and implementation models.

**CPSC 4550\* Computational Science 3 (3)** Introduction to the methods and problems of computational science. Uses problems from engineering and science to develop mathematical and computational solutions. Case studies use techniques from Grand Challenge problems. Emphasizes the use of networking, group development, and modern programming environments.

**CPSC 4620\* Database Management Systems 3 (3)** Introduction to database/data communications concepts as related to the design of online information systems. Problems involving structuring, creating, maintaining, and accessing multiple-user databases are presented and solutions developed. Comparison of several commercially available teleprocessing monitor and database management systems is made.

**CPSC 4630\* On-line Systems 3 (3)** In-depth study of the design and implementation of transaction processing systems and an introduction to basic communications concepts. A survey of commercially available software and a project using one of the systems are included.

**CPSC 4720\* Software Development Methodology 3 (3)** Advanced topics in software development methodology. Techniques such as chief programmer teams, structured design and structured walkthroughs are discussed and used in a major project. Emphasizes the application of these techniques to large-scale software implementation projects. Also includes additional topics such as mathematical foundations of structured programming and verification techniques.

**CPSC (ECE) 4780\* General Purpose Computation on Graphical Processing Units 3 (3)** Instruction in the design and implementation of highly parallel, GPU-based solutions to computationally intensive problems from a variety of disciplines. The OpenCL language with inter-operable OpenGL components is used. Applications to models of physical systems are discussed in detail.

**CPSC 4810\* Selected Topics 1-3 (1-3)** Areas of computer science in which nonstandard problems arise. Innovative approaches to problem solutions which draw from a variety of support courses are developed and implemented. Emphasizes independent study and projects. May be repeated for a maximum of six credits, but only if different topics are covered. Includes Honors sections.

**CPSC 4820\* Special Topics in Computing 3 (3)** In-depth treatment of topics not fully covered in regular courses. Topics vary from semester to semester. May be repeated, but only if different topics are covered.

**CPSC 4910/4911 Seminar in Professional Issues II 3 (2,2)** Considers the impact of computing system development on society. Discusses ethical issues in the design and development of computer software. Students discuss standards for professional behavior, the professional's responsibility to the profession, and techniques for maintaining currency in a dynamic field.

**CPSC 4950 Senior Thesis Research (Honors) 1-3 (1-3)** Directed individual research project for honors students supervised by departmental faculty. May be repeated for a maximum of six credits.



## Appendix 2 Graduate Programs

The school offers four graduate degree programs (PhD/CS, PhD/HCC, MFA/DPA, MS/CS) each containing a strong and common technical core that emphasizes the specialized developed of research and technical skills. Please visit each programs webpage for more information:

- [PhD in Computer Science \(PhD CS\)](#)
- [PhD in Human-Centered Computing \(PhD HCC\)](#)
- [MS in Computer Science \(MS CS\)](#)
- [MFA in Digital Production Arts \(MFA DPA\)](#)

## Appendix 3 Research areas

Algorithms and Data Structures

Animation

Big Data

Bioinformatics

Cloud Computing

Combinatorial Optimization

Computational Social Science

Computer Architecture

Computer Graphics

Computer Science Education

Data Mining and Machine Learning

Databases

Eye Tracking

Games

Graph Theory

Healthcare Informatics

High Performance Computing

Human Computer Interaction

Information Retrieval

Intelligent Virtual Agents

Mobile and Pervasive Computing

Motion Capture

Multimedia Systems

Networking

Operating Systems

Parallel and Distributed Computing

Perception

Programming Languages

Robotics

Scientific Computing

Security and Privacy

Sensor Networks

Software Architecture

Software Engineering

Software Verification and Formal Methods

User-centric Aspects of Recommender Systems

Virtual Reality and Environments

Visualization