

Computer Science

Faculty: Knox Chair; Komagata, , Martinovic, Neff, Sampath, Wolz

Faculty from mathematics with joint teaching appointments in computer science: Conjura, Greenbaun, Iannone

The computer science curriculum is designed to prepare students for employment as computer science specialists, and provide a strong background for advanced study. The degree offered is a BS in Computer Science. All students take courses in programming fundamentals, software engineering, data structures, operating systems, compilers, and computer organization and algorithm analysis. Upper-level options provide exposure to a range of subdisciplines including, but not limited to artificial intelligence, databases, graphics, networks and programming languages. Students are also required to participate in a “practicum” experience in research or industry. All students also take extensive course work in mathematics and science as well as a broad base in arts, humanities, history, and the social sciences.

Requirements for the Major (updated 11/13/03)

I. Required Core Courses

EITHER:

CSC or CMSC 220/CS I: Computational Problem Solving
and

CSC or CMSC 230/CS II: Data Structures and Algorithms
or

CMSC 250/Accelerated CS 1, 2

REMAINING REQUIRED COURSES

CMSC 210/Discrete Structures of Computer Science

CMSC 325/Computer Architecture

CMSC 330/Operating Systems

CMSC 340/CS III: Programming in the Large

CMSC 410/Advanced Analysis of Algorithms

CMSC 434/Compilers and Interpreters

II. Computer Science Options

Select 18 credits from the following list. Students who take CMSC 250 must select 21 credits. Students may take additional options courses for free elective credit.

PART A: Choose nine credits from the following:

CMSC 350/Digital Computer Graphics

CMSC 360/Networks

CMSC 370/Stack Machines

CMSC 380/Artificial Intelligence

CMSC 390/Programming Languages

CMSC 446/Database Management Systems

CMSC 485/Topics in Computer Science

PART B: Choose six credits from the following:

CMSC 493/Internship II in Computer Science

CMSC 497/Research/Presentation Seminar in Computer Science

PART C: Choose three credits from either Part A or B or:

CMSC 499/Independent Study in Computer Science

III. Required Mathematics Courses

MAT or MATH 127/Calculus A

MAT or MATH 128/Calculus B

MAT or MATH 205/Linear Algebra

STA or STAT 215/Statistical Inference

IV. Computer Science Natural Science Options

One laboratory course beyond the general education science requirement either in the same science or in another science. Consult the department for details.

Program Entrance, Retention, and Exit Standards

Every major program at the College has set standards for allowing students to remain in that program, to transfer within the College from one program to another, and to graduate from a program. The following are the standards for computer science programs. Minimum grades are noted in parentheses:

- Retention in the program is based on the following performance standards in these “critical content courses”:
CMSC 210/Discrete Structures of Computer Science (C); CSC or CMSC 220/Computer Science I: Computational Problem Solving (C); CSC or CMSC 230/Computer Science II: Data Structures (C).
- Transfer into the program from another program within the College is based upon the following performance standards in these “foundation courses”:
MAT or MATH 127/Calculus A (C); CSC or CMSC 220/Computer Science I: Computational Problem Solving (C-).
- Graduation requires a GPA of 2.0 in computer science overall and a grade of C in the following courses:
CMSC 210/Discrete Structures of Computer Science; CSC or CMSC 220/Computer Science I: Computational Problem Solving; CSC or CMSC 230/Computer Science II: Data Structures; CMSC 340/Computer Science III: Programming in the Large.

Computer Science Minor (updated 11/13/03)

I. Required Courses

CMSC	210/Discrete Structures of Computer Science
CSC or CMSC	220/CS I: Computational Problem Solving
CSC or CMSC	230/CS II: Data Structures and Algorithms
CMSC	340/CS III: Programming in the Large

II. Options for Computer Science Minor

An additional six credits chosen from the following:

CMSC 325, CMSC 330, CMSC 350, CMSC 360, CMSC 370, CMSC 380, CMSC 390, CMSC 410, CMSC 434, CMSC 446, CMSC 497, CMSC 499.

Minimum grade point average for the minor is the same as for the major.

Department Academic Regulations

A minimum of 21 credits in the major must be earned in the department. A minimum of 15 of the final 21 credits in the major must be earned in the department.

Advanced Placement—If a student has a strong background in a particular major course, then he/she may achieve advanced placement in one of three ways: 1) credit by examination; 2) waiver of the course; or 3) Advanced Placement Credit. Students given permission to waive a course are required to replace it with an upper-level (300 or 400) course.

CSC or CMSC 215, IDSC 105, and HONR 280 may not be taken by computer science majors except with special permission of the department and then only as free electives.

Students who take CMSC 250 accelerate requirements through their junior year.

Suggested Course Sequence

During academic year 2003-2004, The College of New Jersey is in the process of transformative curricular change. Therefore only the course of study for the first-year students entering in 2003-2004 is set out below. These students should consult their advisors when planning courses for future years. Supplements to this online bulletin also will be available on an ongoing basis.

First Year Suggested Sequence CSCA

CMSC	099/Department Orientation Seminar	0
CSC	220/CS I: Computational Problem Solving	4
CSC	230/CS II: Data Structures	4
MAT	127/Calculus A	4
MAT	128/Calculus B	4
WRI	102/Academic Writing	4
IDSC	151/Athens to New York	3 or 4
<i>or</i>		
FSP	101/First Year Seminar	
Foreign Language		6
General Education Electives		3
Total for year		32- 33

All computer science courses require prior completion of the College Basic Skills Requirements in mathematics, reading, and writing.

COURSES (updated 11/13/03)

CSC 101 Introduction to Interactive Computing 4 cr

(4 class hours)

Prerequisites: none

(every semester)

Co-listed with Interactive Multimedia Program IMM 120

A first course in computing languages for interactive multimedia. Students are introduced to the art of programming through state-of-the-art multimedia technologies (e.g. Macromedia Studio MX). Through intensive laboratory experience students learn the programming fundamentals (e.g. variables, functions, control structures and logic, persistent storage, and networking). Problems related to interactivity are emphasized (e.g. through assignments based on HTML and Flash coding). Students will understand the distinction between markup languages, scripting languages, and general purpose programming languages and develop proficiency in the first two. Persistent storage and networking concepts are introduced through high-level applications (e.g. Macromedia Studio). Efficiency, data structure organization, and objects are introduced within the context of interactive computing problem solving. **This course may not be taken by computer science majors without permission of the department, and then only for free elective credit.**

CMSC 210/Discrete Structures of Computer Science 3 cr.

(3 class hours)

(every semester)

Prerequisite: MAT or MATH 127

Concepts and structures fundamental to computer science. Declarative programming techniques will be used to explore discrete structures. Topics will include logic, relations, functions, word algebras, induction, and recursion.

CSC 215/Computer Science I for Science and Engineering 3 cr.

(3 class hours)

(every semester)

A first course in computer science for science, mathematics, and engineering science majors. Emphasis is on using computational methods to solve scientific problems. The programming languages C/C++ or Fortran will be used to teach structured programming and algorithm development. Topics include control structures, data typing including structures and arrays, parameterized procedures, and recursion as well as simple I/O control. **This course may not be taken by computer science majors without permission of the department, and then only for free elective credit.**

CSC 220/Computer Science I: Computational Problem Solving 4 cr.

(4 class hours)

(every semester)

A first course in computer science for computer science majors in which students learn to express algorithmic ideas in an abstract manner. An object-oriented language such as Java will be introduced; however, the emphasis is on algorithmic design and implementation rather than language mechanics. Students will become intimately familiar with expressing mathematical ideas as programs.

CSC 230/Computer Science II: Data Structures 4 cr.

(4 class hours)

(every semester)

Prerequisite: CSC or CMSC 220 with a grade of C or higher

A second course in computer science in which students learn how the algorithms and structures studied in CMSC 220 are implemented on a sequential machine. Classic data structures (lists, queues, stacks, trees, and tables) and algorithms (searching and sorting) are considered, but the emphasis is on analysis. First analysis of implementation techniques is addressed by studying the trade-offs between static and dynamic structures. Second analysis of complexity of algorithms is studied first informally and then using formal proof techniques. The implementation of the object-oriented paradigm is also shown via Java.

CMSC 250/Accelerated CS I, II 4 cr.

(4 class hours)

(annually)

Prerequisite: Permission of the department

A first course in computer science for students with extensive experience programming, or those who have placed above a standard set by the department on the CS AP exam. It is also for those with some computer science background who have gaps in their knowledge of the standard CS 1/CS 2 curriculum or took these courses elsewhere but have no experience programming in the programming language or paradigm that supports our CS 1 and 2 sequence. The course covers the material of CMSC 220 (CS 1) and CMSC 230 (CS 2) in one semester. The basic introduction to programming of CMSC 220 is considered a review, while the emphasis on problem solving and solution design is presented within the context of a thorough grounding in the classic data structures using the modern object-oriented framework.

NOTE: Students who take this course must complete a single additional free elective credit as well as an additional three credits of CS options course or receive Advanced Placement credit.

CMSC 325/Computer Architecture **4 cr.**

(4 class hours)

(annually)

Prerequisites: CMSC 210 and CSC or CMSC 230, both with a grade of C or higher

The levels of abstraction in computer hardware are examined in theory and practice. The classic components of a computer system including CPU control and datapath, ALU, memory, and input/output are covered, with supporting case studies in design and simulation. The use of a register transfer language is examined. Hardwired control and micro-programming control are examined in support of the implementation of a machine language. A design project ties together machine organization, assembly language programming, and logic and design fundamentals. In addition to traditional architectures, high-performance computing is considered, including CISC/RISC studies, pipelined architectures, and parallel processors.

CMSC 330/Operating Systems **3 cr.**

(3 class hours)

(annually)

Prerequisites: CMSC 325, CMSC 340, the latter with a grade of C or higher

Study of resource management implemented by an operating system in multiprogramming environment with respect to CPU, memory, file, and device. Emphasis on programming techniques leading to system software design, development, and implementation.

CMSC 340/Computer Science III: Programming in the Large **4 cr.**

(4 class hours)

(annually)

Prerequisite: CSC or CMSC 230 with a grade of C or higher

This course introduces the principles of the design and development of large software systems. Students are expected to apply those principles to the solution and implementation of real-world problems. Students will learn C++, focusing on object-oriented programming techniques. Algorithm complexity will continue to be studied informally. Documentation and product presentation will be expected.

CMSC 350/Digital Computer Graphics **3 cr.**

(3 class hours)

(annually)

Prerequisites: CMSC 210, CMSC 340 both with a grade of C or higher

A thorough grounding in the fundamental principles of digital computer graphics methods for students with both computer science and mathematical sophistication. Emphasis is on algorithms and methods, not on using canned graphics packages. Digital display devices, scan conversion algorithms, windows and viewports, coordinate transformations in two and three dimensions, projections, perspective transformations, clipping, systems of graphics procedures, interactive graphic input, hard copy devices, half-toning, splines and Bezier curves, hidden line and surface removal, and fractals.

CMSC 360/Networks **3 cr.**

(3 class hours)

(annually)

Prerequisites: CMSC 210, CMSC 340, both with a grade of C or higher

This course will introduce concepts of network programming. The emphasis will be on protocols used to communicate between various implementations of UNIX as well as PCs, Macs, and other systems. Topics include: ethernet, token ring, fiber-optic topologies; IP, ICMP, UDP, TCP protocols; applications such as telnet, ftp, ping; ATM networks. Laboratory experiences will include the use of network viewing tools such as traceroute, tcpdump, and dig. Network applications will be written in such languages as C, Perl, and TCL.

CMSC 370/Stack Machines **3 cr.**

(3 class hours)

(occasionally)

Prerequisite: CMSC 325

The application and implementation of virtual stack machines and other intermediate-level models of computation.

CMSC 380/Artificial Intelligence **3 cr.**

(3 class hours)

(annually)

Prerequisites: CMSC 210, CSC or CMSC 230, both with a grade of C or higher

The study of how to make the computer behave intelligently. Topics: state-space methods of problem solving, heuristic search techniques, representation and use of knowledge, applications and design of expert systems, natural language processing, vision and image understanding, and programming in LISP, Prolog, or any of the latest AI languages.

CMSC 390/Programming Languages 3 cr.

(3 class hours)
(annually)

Prerequisites: CMSC 210 and CMSC 340, both with a grade of C or higher

This course covers alternatives to the imperative procedural programming paradigm. Included are object-oriented, functional, and logic programming. The emphasis is on language design, use, and implementation.

CMSC 393/Internship I in Computer Science 3–6 cr.

(every semester)

Prerequisite: Permission of internship coordinator

Counts as a general elective course. A supervised computer-related field experience in industrial, governmental, or private sector.

Faculty supervisor advises the student where to focus the learning objective for the given internship. Grading is pass/unsatisfactory.

Courses graded on a P/U basis are not counted as part of the 64 semester hours of letter-graded courses required for graduation with honors.

CMSC 410/Advanced Analysis of Algorithms 3 cr.

(3 class hours)
(annually)

Prerequisites: CMSC 210 and CMSC 340, both with a grade of C or higher

This course presents the major principles of algorithm design and analysis, and applies those principles to classical problems in computer science. Topics include complexity, advanced ADTs, searching and sorting, graph search and traversal, dynamic programming, theoretical computer science, operations on polynomials and matrices, and pattern matching.

CMSC 434/Compilers and Interpreters 3 cr.

(3 class hours)
(annually)

Prerequisites: CMSC 210 and CMSC 340, both with a grade of C or higher

Theory and practice of designing and implementing lexical analyzers, compilers, interpreters, and macro assemblers. Formal language theory, regular expressions, Backus-Naur form, formal and informal specification of semantics. Deterministic and nondeterministic finite state automata, recursive descent and predictive parsing, and code generation. Issues of programming language design.

CMSC 446/Database Management Systems 3 cr.

(3 class hours)
(annually)

Prerequisites: CMSC 210 and CMSC 340, both with a grade of C or higher

This course introduces the student to abstracts of well-known database systems and the design and implementation of a database system using a relational database package.

CMSC 485/Topics in Computer Science 3 cr.

(3 class hours)
(every semester)

Prerequisites: CMSC 210 and CMSC 340, both with a grade of C or higher

Study of an advanced topic in computer science chosen by the instructor. Normally taken by junior or senior computer science majors, this course may be elected several times, as long as the topics differ.

CMSC 493/Internship II in Computer Science 3–6 cr.

(every semester)

Prerequisite: Permission of instructor

A supervised computer-related field experience in industrial, governmental, or private sector. Faculty supervisor advises the student where to focus the learning objective for the given internship. A project at the internship is to be the level of research in junior/senior independent study topics in computer science. Grading is a letter grade. Paper and public lecture required.

CMSC 497/Mentored Research in Computer Science 3 cr.

(3 class hours)
(every semester)

Prerequisite: Permission of instructor

Intensive study of an advanced topic in computer science under the supervision of a faculty mentor. Emphasizes student activity, use of journals and monographs, discussions, solution and presentations of problems. This course culminates in the writing of a journal-style paper and the presentation of a public lecture.

CMSC 499/Independent Study in Computer Science 3 cr.

(every semester)

Prerequisite: By invitation only

Student will study independently an appropriate area. A faculty adviser and a project description must be agreed upon before registering for this course.
