Preteem Ghosh, Ph.D.
Professor and interim chair

egr.vcu.edu/departments/computer

The Department of Computer Science offers undergraduate and graduate programs. The Bachelor of Science in Computer Science is a rigorous, highly concentrated curriculum of computer science courses. It includes advanced study in several important areas of computer science and provides a strong foundation in this discipline. Every course is taught by full-time faculty members who also serve as advisers to both undergraduate and graduate students.

The master’s degree program emphasizes continuing self-development of individuals currently engaged in science-, technology- and engineering-related fields. It prepares persons who have completed undergraduate majors in these fields for entry into careers in areas that use computing technology. Both the theoretical and applied aspects of computer science are emphasized in this program. The program offers courses in a wide range of areas in computer science, including machine learning, artificial intelligence, cybersecurity and cloud computing, data mining, bioinformatics, and medical informatics.

- Computer Science, Bachelor of Science (B.S.) (http://bulletin.vcu.edu/undergraduate/engineering/computer-science/computer-science-bs/)
- Computer Science, Bachelor of Science (B.S.) with a concentration in cybersecurity (http://bulletin.vcu.edu/undergraduate/engineering/computer-science/computer-science-bs-concentration-cybersecurity/)
- Computer Science, Bachelor of Science (B.S.) with a concentration in data science (http://bulletin.vcu.edu/undergraduate/engineering/computer-science/computer-science-bs-concentration-data-science/)
- Computer Science, Bachelor of Science (B.S.) with a concentration in software engineering (http://bulletin.vcu.edu/undergraduate/engineering/computer-science/computer-science-bs-concentration-software-engineering/)
- Computer Science, minor in (http://bulletin.vcu.edu/undergraduate/engineering/computer-science/computer-science-minor/)
- Computer Science, Certificate in (Post-baccalaureate undergraduate certificate) (http://bulletin.vcu.edu/undergraduate/engineering/computer-science/computer-science-certificate/)
- Fundamentals of Computing (Baccalaureate certificate) (http://bulletin.vcu.edu/undergraduate/engineering/computer-science/fundamentals-computing-certificate/)

CMSC 101. Introduction to Computer Science. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: MATH 139 or MATH 141 with a minimum grade of C. An introduction to the work of computer scientists, including an overview of current research and application areas as well as career opportunities. Topics include problem-solving, the basics of computer organization, the software engineering life cycle, research resources and social and ethical aspects of technology. Additional topics also include binary, hexadecimal, two’s complement, floating point representation, ASCII and Unicode.

CMSC 144. Code Beats With Python. 1 Hour.
Semester course; 2 laboratory hours. 1 credit. An introduction to computer programming in Python by teaching students to create hip hop beats. Teaches fundamental programming concepts including sequencing, syntax, variables, functions, parameters, lists, repetition and modularization. Teaches just enough music theory to ensure that student-made beats sound great, including fundamental concepts such as melody, rhythm, harmony, chord progression and orchestration. Students will complete in-class activities that reinforce class concepts and, if completed correctly, demonstrate a clear understanding of the material.

CMSC 191. Topics in Computer Science. 3 Hours.
Semester course; 3 lecture hours. 3 credits. May be repeated for credit. Prerequisite: permission of the instructor. This course will teach selected topics in computer science. See the Schedule of Classes for specific topics to be offered each semester and prerequisites.

CMSC 210. Computers and Programming. 3 Hours.
Semester course; 3 lecture hours (delivered online). 3 credits. Introduction to object-oriented programming using Python. The course introduces students to structured programming logic and design techniques. The course content also includes instruction in critical thinking and problem-solving skills using contemporary tools. Specific topics include flowcharting, pseudocode and program control structures, including sequence, selection and repetition. This course is not applicable for credit toward the B.S. in Computer Science.

CMSC 235. Computing and Data Ethics. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Introduction to ethical considerations for computing and working with data, legal implications surrounding data and computational systems, and sociotechnical responsibilities.

CMSC 245. Introduction to Programming Using C++. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: MATH 151 or satisfactory score on the Mathematical Placement Test. Students registering for CMSC 245 must have taken the VCU Mathematics Placement Test within the one-year period immediately preceding the beginning of the course. An exception to this policy is made in the case in which the stated alternative prerequisite course has been completed at VCU. Students are expected to have fundamental computer skills. Introduction to the concepts and practice of structured programming using C++. Problem-solving, top-down design of algorithms, objects, basic C++ syntax, control structures, functions and arrays. This course is intended for engineering majors.

CMSC 246. Advanced Programming Using C++. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 245. Advanced programming in C++. Topics include program design, objects, classes, inheritance, files, strings, linked lists, stacks, queues, binary trees, recursion, and basic searching and sorting techniques. This course is intended for engineering majors.

CMSC 254. Introduction to Problem-solving. 4 Hours.
Semester course; 3 lecture and 2 laboratory hours. 4 credits. Introduction to problem solving and implementation of solutions using Python. The course introduces students to concepts and practice of structured programming, problem-solving, top-down design of algorithms, a Python language syntax, control structures and arrays. The course content also includes instruction in critical-thinking and problem-solving skills using contemporary tools. Specific topics include flowcharting, pseudocode and program control structures, including sequence, selection, repetition and modularization.
CMSC 255. Introduction to Programming. 4 Hours.
Semester course; 3 lecture and 2 laboratory hours. 4 credits. Prerequisite: calculus-level placement on the VCU Mathematics Placement Test within the one-year period immediately preceding enrollment in the course, or MATH 151 or equivalent. Students are expected to have fundamental computer skills. Introduction to object-oriented programming using Java. Topics include problem-solving, top-down design of algorithms using control structures, methods, arrays, basic I/O, basic concepts of objects and classes in Java, Java classes for manipulating strings, and introduction to program testing, UML notation and integrated development environments. Students may not receive credit for both CMSC 255 and INFO 250.

CMSC 256. Data Structures and Object Oriented Programming. 4 Hours.
Semester course; 3 lecture and 2 laboratory hours. 4 credits. Prerequisite: CMSC 255 with a minimum grade of C, corequisite: CMSC 302. Advanced programming using Java. Topics include introduction to object-oriented design, inheritance, polymorphism, exceptions, interfaces, linked lists, stacks, queues, binary trees, recursion, and basic searching and sorting techniques. Continued focus on program testing and UML notation. Students may not receive credit for both CMSC 256 and INFO 350.

CMSC 257. Computer Systems. 4 Hours.
Semester course; 3 lecture and 2 laboratory hours. 4 credits. Prerequisite: CMSC 256 with a minimum grade of C. Topics include UNIX essentials; system programming in C; machine-level representation and organization of programs/data, arrays and pointers; types, structs and unions; strings; bit/byte operations; memory management; shell programming; input/output, including file handling; debugging; signals; network programming using sockets; program concurrency using forks and threads; experiments on program performance and optimization techniques.

CMSC 302. Introduction to Discrete Structures. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 255 with minimum grade of C. Logic and proofs, sets, functions, sequences and sums, relations, graphs, trees, induction and recursion, advanced counting technique (recurrences).

CMSC 303. Introduction to the Theory of Computation. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 302 or the equivalent with a grade of C or better. Complexity classes, grammars, automata, formal languages, Turing machines, computability.

CMSC 311. Computer Organization. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 302 with minimum grade of C; corequisite: CMSC 257. Introduction to the basic organization of computers including elementary digital logic design, processor and arithmetic/logic unit design, data paths, memory hierarchy, I/O devices, instruction set architecture and addressing modes.

CMSC 312. Introduction to Operating Systems. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 311 or EGRE 364. Computer systems design, I/O processing, secondary memory organization, command languages, memory management and job scheduling. Students will work in teams to design and implement an operating system simulation.

CMSC 320. Software Engineering and Web Development. 3 Hours.
Semester course; 3 lecture hours (delivered online). 3 credits. Prerequisite: CMSC 210 or CMSC 254. Introduction to software engineering and web development. The course introduces students to the software development process, including design, development and testing principles. Students will apply these principles in the development of a web application. This course is not applicable for credit toward any College of Engineering degrees.

CMSC 330. Data Science Skills. 3 Hours.
Semester course; 3 lecture hours (delivered online). 3 credits. Prerequisite: CMSC 210 or CMSC 254. Introduction to data science skills. The course introduces students to the foundations of data science and the tools used to collect, analyze and represent data. Students will apply these principles in both analysis and visualization projects. This course is not applicable for credit toward any College of Engineering degrees.

CMSC 340. Cybersecurity Skills. 3 Hours.
Semester course; 3 lecture hours (delivered online). 3 credits. Introduction to cybersecurity skills. The course introduces students to cybersecurity terminology, standards and best practices. Students will apply these practices as part of a cybersecurity-focused project. This course is not applicable for credit toward any College of Engineering degrees.

CMSC 355. Fundamentals of Software Engineering. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 256 or EGRE 246, either with a minimum grade of C. Provides an overview of how to engineer software systems, including all stages of the software development process based on agile principles. Familiarizes students with modern software tooling and the principles of software quality and testing. Students will work in teams to gain experience in software development methodology, write specification and design documents, and develop a prototype.

CMSC 391. Topics in Computer Science. 3 Hours.
Semester course; 3 lecture hours. 3 credits. May be repeated for credit. Prerequisite: permission of the instructor. This course will teach selected topics in computer science. See the Schedule of Classes for specific topics to be offered each semester and prerequisites.

CMSC 401. Algorithm Analysis with Advanced Data Structures. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 256 with a grade of C or better and CMSC 302 with a grade of C or better. Introduction to algorithm analysis and complexity classes. Advanced data structures topics including multiple linked lists, height-balanced trees, hashing and graph representation; incorporating data structures into object-oriented design. Analysis of various searching and sorting algorithms. Algorithm design topics include divide-and-conquer, dynamic programming and greedy methods.

CMSC 403. Programming Languages. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 256 and CMSC 303, both with a minimum grade of C. Survey of representative modern programming languages. Formal definition of programming languages including specifications of syntax and semantics. Precedence, infix, prefix and postfix notation. Global properties of algorithmic languages. Sub-routines, co-routines and tasks. List processing, string manipulation, data description and simulation languages. Run-time representation of program and data structures.

CMSC 404. Compiler Construction. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 401 and 403. A first course in compiler theory and construction. Formal description of languages, underlying theory and design techniques for compilers, lexical analysis, syntax analysis, syntax-directed translation, intermediate languages, run-time system management, code generation, code optimization, compiler-building tools.

CMSC 409. Artificial Intelligence. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 401 with a minimum grade of C and MATH 310. Covers problem spaces, problem-solving methods, game playing, knowledge representation, expert systems, natural language understanding.
CMSC 410. Introduction to Quantum Computing. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 401 and MATH 310, both with a minimum grade of B. Introduction to quantum information processing: state vectors and density operators, tensor product space, unitary evolution, no-go theorems, measurement, qubit, gate model of quantum computing, quantum complexity theory, quantum error correction, quantum algorithms, and quantum machine learning.

CMSC 411. Computer Graphics. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 355 and MATH 310. Presents mathematical techniques for graphic development and transformation, curve and surface approximation and projections, graphical languages and data structures and their implementation, graphic modeling.

CMSC 412. Social Network Analysis and Cybersecurity Risks. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisites: CMSC 401 with a minimum grade of C. Covers network models, link prediction and analysis, centrality measures, random networks, power-laws and preferential attachment, small world phenomenon and decentralized search, community structure, information propagation in networks, and security and privacy issues in OSNs.

CMSC 413. Introduction to Cybersecurity. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 with a minimum grade of C. This course provides introduction and basic concepts of computer security, cyber attacks, cyber defense, cyber forensics and cyber ethics.

CMSC 414. Computer and Network Security. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 with a minimum grade of C. Corequisite: CMSC 312. This course covers the best practices of computer systems and network security. Key topics include security architecture, cryptographic systems and security management tools.

CMSC 415. Introduction to Cryptography. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 with a minimum grade of C. This course provides a rigorous and theoretical introduction to modern cryptography. Key topics include symmetric key encryption and authentication, public key encryption, and digital signatures.

CMSC 416. Introduction to Natural Language Processing. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 with a minimum grade of C. Covers rule-based and statistical methods for creating computer programs that analyze, generate and understand human language. Topics include regular expressions and automata, context-free grammars, probabilistic classifiers, and machine learning. Word-level, syntactic and semantic processing are all considered. Application to real-world problems such as spell-checking, Web search, automatic question answering, authorship identification and developing conversational interfaces.

CMSC 420. Software Project Management. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 355 with a minimum grade of C. Study of the logistics of team software development. Students work in teams to gain experience in software management and develop the components of a larger software product. Topics include risk management, project planning, quality management, configuration management and software testing.

CMSC 425. Introduction to Software Analysis and Testing. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 355 with a minimum grade of C. Enrollment is restricted to majors in the computer science program. A practical introduction to testing complex software applications. An introduction to concepts and techniques used in the analysis of software, including basic and advanced control flow and data flow analyses. Using analytic results to derive test data and validate the correct implementation of programs. Advanced testing strategies including random, structural, mutation and fuzzing.

CMSC 428. Mobile Programming: iOS. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 355, with a minimum grade of C. This course covers the fundamentals of Swift, Xcode and iOS for programming and design of iOS applications. Background in object-oriented programming and access to a computer with Xcode platform is required.

CMSC 435. Introduction to Data Science. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 401 with a minimum grade of C. This course covers understanding, representation, storage, retrieval, preprocessing and analysis of data. Specific topics include data quality and preprocessing, database management systems, data warehouses, selected methods for scalable unsupervised and supervised data analysis, and assessment of results generated by these methods. Students will be engaged in analysis of real-life data from data pre-processing, through data analysis, to the assessment of a knowledge product.

CMSC 440. Data Communication and Networking. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 257 with a minimum grade of C. Enrollment is restricted to majors in the College of Engineering. This course explores computer networking, focusing on the applications and protocols that run on the Internet. Students will take a top-down approach to the layered network architecture, studying applications first and then proceeding down the network "stack" toward the physical link. Students will examine the operation of applications such as the web, FTP, e-mail and DNS. At the transport layer, students will study both connectionless UDP and connection-oriented TCP, with an in-depth study of TCP operation, specifically flow control and congestion control. Data communications are explored through various data routing protocols. Additional topics include network security and wireless/mobile networking.

CMSC 441. Senior Design Studio I (Laboratory/Project Time). 2 Hours.
Semester course; 6 laboratory hours. 2 credits. Prerequisites: CMSC 355; and UNIV 200 or HONR 200 or equivalent, both with minimum grades of C. Corequisite: CMSC 451. Enrollment is restricted to computer science majors with senior standing who have 24 credits in computer science courses. A minimum of six laboratory hours per week dedicated to the execution phase of the senior design (capstone) project for the computer science major. Tasks include team meetings, brainstorming, sponsor advising, researching, designing, implementing, reviewing, testing and validating projects. Each student will participate, either individually or as part of a team, in a project or other experience approved by the course coordinator or sponsored by another computer science faculty member. Students will submit a detailed written description of their proposed project or experience and will present orally some aspect of what they have learned and/or done during the semester. This course cannot be counted as upper-level CMSC electives for students graduating under bulletins prior to 2008-09.
CMSC 442. Senior Design Studio II (Laboratory/Project Time). 2 Hours.
Semester course; 6 laboratory hours. 2 credits. Prerequisites: CMSC 441, CMSC 451 and CMSC 508, each with a minimum grade of C. Corequisite: CMSC 452. Enrollment is restricted to computer science majors with senior standing who have 24 credits in computer science courses. A minimum of six laboratory hours per week dedicated to the execution phase of the senior design (capstone) project for the computer science major. Tasks include team meetings, brainstorming, sponsor advising, researching, designing, implementing, reviewing, testing and validating projects. Each student will participate, either individually or as part of a team, in a project or other experience approved by the course coordinator or sponsored by another computer science faculty member. Students must continue on the same project that was started in CMSC 441 and CMSC 451. A final project report and presentation are due at the conclusion of the two-semester project or experience. This course cannot be counted as upper-level CMSC electives for students graduating under bulletins prior to 2008-09.

CMSC 451. Senior Project I. 1 Hour.
Semester course; 1 lecture hour. 1 credit. Prerequisites: CMSC 355 with minimum grade of C; and UNIV 200 or HONR 200 or equivalent. Corequisite: CMSC 441. Enrollment is restricted to computer science majors with senior standing who have 24 credits in computer science courses. This weekly seminar presents and discusses topics relevant to senior-level computer science students in support of the capstone project and upcoming graduation. A single course coordinator manages and administers the course and schedules the various faculty lectures and guest speakers. Topics include, but are not limited to, the following: proposal writing; project planning and management; scheduling resources and budgeting for software development projects; patents and intellectual property; entrepreneurship; ethical, legal and social issues in computing; and professional responsibilities of computer scientists. Each student will write and revise a research paper on a technical topic associated with his or her project or experience. A final project report and presentation, which will include a discussion of associated legal, social and/or ethical issues, are due at the conclusion of the two-semester project or experience. The courses in this sequence cannot be counted as upper-level CMSC electives for students graduating under bulletins prior to 2008-09.

CMSC 452. Senior Project II. 1 Hour.
Semester course; 1 lecture hour. 1 credit. Prerequisites: CMSC 441, CMSC 451 and CMSC 508, each with a minimum grade of C. Corequisite: CMSC 442. Enrollment is restricted to students with senior standing in the computer science department. This weekly seminar presents and discusses topics relevant to senior-level computer science students in support of the capstone project and upcoming graduation. A single course coordinator manages and administers the course and schedules the various faculty lectures and guest speakers. Topics include, but are not limited to, the following: proposal writing; project planning and management; scheduling resources and budgeting for software development projects; patents and intellectual property; entrepreneurship; ethical, legal and social issues in computing; and professional responsibilities of computer scientists. Each student will write and revise a research paper on a technical topic associated with his or her project or experience. Students must continue on the same project that was started in CMSC 451. A final project report and presentation, which will include a discussion of associated legal, social and/or ethical issues, are due at the conclusion of the two-semester project or experience. The courses in this sequence cannot be counted as upper-level CMSC electives for students graduating under bulletins prior to 2008-09.

CMSC 455. Software as a Service. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 355 with a minimum grade of C. Enrollment is restricted to majors in the computer science program. Students will examine the challenges, opportunities and open problems of software-as-a-service deployed on commodity cloud computing platforms. Covers relevant software architectures and API design principles. Includes concepts of modern software frameworks for software development, cloud computing for software deployment and software operations. Students participate in projects that use modern tooling to develop, deploy and monitor a software application.

CMSC 475. Design and Implementation of User Interfaces. 3 Hours.
Semester course; 3 lecture hours. 3 credits. Prerequisite: CMSC 355 with a minimum grade of C. Enrollment is restricted to majors in the computer science program. This course investigates the design and implementation of user interfaces and the evaluation of user experiences. Particular emphasis is placed on creating professional-quality designs and implementations and on evaluating these implementations with end-users. Students will create their own UIs as well as critique others to develop a deep understanding of what works in practice.

CMSC 491. Topics in Computer Science. 1-3 Hours.
Semester course; variable hours. 1-3 credits. May be repeated for credit with different content. Prerequisite: permission of instructor. This course will cover selected topics in computer science. See the Schedule of Classes for specific topics to be offered each semester.

CMSC 492. Independent Study. 2-4 Hours.
Semester course; variable hours. 2, 3 or 4 credits per semester. Maximum 4 credits per semester; maximum total of 6 credits. Generally open only to students of junior or senior standing who have acquired at least 12 credits in the departmental discipline. Determination of the amount of credit and permission of instructor and department chair must be procured prior to registration of the course. The student must submit a proposal for investigating some area or problem not contained in the regular curriculum. The results of the student’s study will be presented in a report.