GEORGE MASON UNIVERSITY

Volgenau School of Information Technology and Engineering

COMPUTER SCIENCE, B.S.

2008-2009

The objectives of the B.S. program in Computer Science relate to the abilities of the graduates several years after graduation. The objectives include

- Foundation for successful careers in industry: Graduates of the program will have a broad understanding of the fundamental concepts, methodologies and tools, and applications of computer science. They will have the educational foundation that leads to successful careers in the computing industry.
- Foundation for graduate study: Graduates of the program will have the academic preparation for successful completion of rigorous graduate programs.
- Professional preparation: Graduates will be effective in written, oral, and visual communication, and able to work collaboratively with others in a professional and ethical manner.

This bachelor's degree program is accredited by the Computing Accreditation Commission of the Accreditation Board for Engineering and Technology (ABET), 111 Market Place, Suite 1050, Baltimore Maryland 21202-4012.

Admission Requirements

Admission to George Mason is competitive. Each candidate who presents sufficient admission qualifications is reviewed in the context of other qualified applicants. An offer of admission is valid only for the semester for which the student applied. Application for undergraduate admission should be made to the Office of Admissions. Applications are available at http://admissions.gmu.edu/applynow. (See the last page for details about admission to the BS/Accelerated MS programs.)

Freshman Admission Requirements

The following factors are considered when reviewing applications for admission:

- Cumulative high school grade point average for course work completed in grades 9 through 12.
- Level of difficulty of course work elected throughout the high school years, particularly in English, mathematics, laboratory science, social science, and foreign language.
- Scores from Scholastic Aptitude Test I (SAT I) or the American College Test (ACT).
- Test of English as a Foreign Language (TOEFL) where applicable.
- Essays, list of extracurricular activities, and teacher and guidance counselor recommendations.

Transfer Admission Requirements

The university accepts qualified students who wish to transfer from other colleges. Transfer applicants must submit two copies of official transcripts from each collegiate institution attended. Transfer applicants with fewer than 30 semester hours of transferable credit must also submit a copy of their secondary school record and test scores. All non-native English speakers are also required to submit a TOEFL or IELTS score or acceptable grades (C or better) in at least two English composition or literature classes taken at another U.S. university or college.

Undecided Students

Students who are undecided but interested in pursuing a career in Computer Science should seek the advice of a departmental faculty advisor. Sample schedules of the majors within the Volgenau School of Information Technology and Engineering are also available from each individual department.

Change of Major

Students requesting a change of major to computer science must have a GPA of at least 2.75 and successfully completed two of the following: CS 112, 211; MATH 113, 114, or 125.

Advanced Placement, Credit by Exam

Some students may receive credit for CS 112 or 211 by passing departmentally administered exams. In addition, a score of 3 on the Advanced Placement (AP) computer science exam qualifies the student for credit in CS 112. An AP score of 4, together with demonstrated competence in the programming language used in CS 211, qualifies students for credit in CS 211. A score of 4 on the International Baccalaureate (IB) computer science exam qualifies students for credit in CS 112, and a score of 5 or more qualifies students for credit in CS 211.

Degree Requirements

Undergraduate degree work in computer science provides students with essential background for studying the design and implementation of computer system software, computer architecture, and computer software applications for science and business. The program emphasizes both computer system fundamentals and computer software applications. Required areas of study include data structures, analysis of algorithms, assembly language, computer architecture and language translation, software design and development, artificial intelligence and networking. Evolving software technologies are a major concern. The BS in Computer Science program also requires 23 credits in mathematics and statistics, including calculus, discrete mathematics, matrix algebra, applied probability theory, and numerical analysis. A sample schedule that fulfills degree requirements for a Bachelor of Science in Computer Science degree is shown below. With prior approval of department advisors, some courses may be taken out of the indicated sequences, particularly in the case of General Education [GE] courses.

Sample Schedule for B.S. in Computer Science

First Semester	
CS 101 Preview of Computer Science	2
CS 112 Intro Computer Programming	4
MATH 113 Calculus I	4
ENGL 101 Composition [GE]	3
Western Civilization [GE]	3
Total Hours	16

Second Semester	
CS 211 Object-Oriented Programming	3
MATH 114 Calculus II	4
Fine Arts [GE]	3
Literature [GE]	3
CS105 Computer Ethics	1
Total Hours	14

Third Semester	
ECE 301 Digital Electronics	3
MATH 213 Calculus III	3
Social and Behavioral Science [GE]	3
Natural Science with lab	4
CS 262 Intro Low-level Programming	1
Total Hours	14

Fourth Semester	
CS 310 Data Structures	3
MATH 125 Discrete Mathematics	3
Elective	3
Natural Science with lab	4
COMM 100 Oral Communication [GE]	3
Total Hours	16

Fifth Semester	
CS 330 Formal Methods & Models	3
CS 367 Computer Systems & Programming	3
MATH 203 Matrix Algebra	3
ENGL 302 Advanced Composition [GE]	3
Global Understanding [GE]	3
Total Hours	15

Sixth Semester	
CS 465 Computer Architecture	3
CS 421 Software Requirements and Design	3
STAT 344 Probability and Statistics	3
Natural Science with lab	4
Humanities	3
Total Hours	16

Seventh Semester	
CS 483 Analysis of Algorithms	3
OR 481 Numerical Methods	3
Senior CS Course	3
Senior CS Course	3
CS-related Elective	3
Total Hours	15

Eighth Semester	
CS 306 Synthesis - Ethics & Law [GE]	3
Senior CS course	3
Senior CS course	3
CS-related Elective	3
Elective	2
Total Hours	14

B.S. DEGREE IN COMPUTER SCIENCE
FROM 2008-2009 CATALOG
(The catalog is the official reference)

GENERAL EDUCATION [GE] REQUIREMENTS (24 credits)

FOUNDATION
- ✓ ENGL 101 Composition [Non-native English speakers see catalog about using ENGL 100]
- ✓ ENGL 302 Advanced Composition (Natural Science section)
- ✓ COMM 100 Oral Communication
- ✓ [Information Technology is satisfied by major requirements]
- ✓ [Quantitative Reasoning is satisfied by major requirements]

CORE [see university catalog for currently approved classes for these requirements]
- ✓ Literature
- ✓ Western Civilization
- ✓ Social and Behavioral Science
- ✓ Global Understanding
- ✓ Fine Arts
- ✓ [Synthesis is satisfied by the major requirement - CS 306]
- ✓ [Natural Science is satisfied by major requirements]

MAJOR REQUIREMENTS (91 credits)

Required Computer Science
- ✓ CS 101
- ✓ CS 105
- ✓ CS 112
- ✓ CS 211
- ✓ CS 262
- ✓ CS 306
- ✓ CS 310
- ✓ CS 330
- ✓ CS 367
- ✓ CS 421
- ✓ CS 465
- ✓ CS 483
- ✓ **Senior CS** -- any four of the following:
  CS 440, 450, 451, 455, 468, 471, 475, 480, 482, 484, 490

Mathematics and Engineering
- ✓ MATH 113, MATH 114, MATH 213
- ✓ MATH 125, MATH 203, STAT 344*
- ✓ OR 481 [or MATH 446]
- ✓ ECE 301

Computer Science Related Electives – Any two of the following (not used to satisfy other requirements):
ECE 280, 431, 442, 447, 450, 511; OR 335, 441, 442; PHIL 371, 376; STAT 354, SWE 432, 437, 443; SYST 371, 470; any MATH course numbered above 300 except MATH 351; any CS course above 300.

Natural Sciences [see next page]

Humanities [see next page]

GENERAL ELECTIVES (5 credits)
[Unrestricted academic hours beyond General Education and Major requirements.]

* Students may replace STAT 344 with both MATH 351 and 352 (while also satisfying one CS-related elective.)

Grade and Credit Hour Requirements for CS Majors

Students must earn a C or better in any course intended to satisfy a prerequisite for a computer science course. Computer science majors may not use more than one course with grade C- or lower toward departmental requirements. (Any course can be repeated and the new grade is used in the computation of the cumulative GPA; see "repeating a course" in the University Catalog.)

Graduation requires 120 total hours (at least 30 at GMU) and 45 upper division hours (at least 12 at GMU).

Humanities Requirement for CS Majors

One additional course (3 credit hours) is required in the Humanities beyond what is required by the General Education [GE] requirements. This Humanities requirement can be fulfilled by any course that could satisfy a "core" GE requirement (except for the synthesis course and the natural sciences).

Natural Science Requirement for CS Majors

The BS in CS requires 12 hours of natural science. The courses should be intended for science and engineering students and must include a two course sequence with laboratories. Some acceptable combinations have a total of more than 12 hours. As with all courses, be sure that you have the prerequisites.

Students who wish to study a natural science in depth can choose to take the coursework that students majoring in that science are required to take; these course are more rigorous. More advanced courses may be substituted with the advisor's approval. Students with a strong interest in Biology should consider the BS in Applied Computer Science.

Approved Two Course Sequences with Laboratories

ASTRONOMY 111(3)/112(1), 113(3)/114(1). The labs are computer-based.

BIOLOGY 103(4), 104(4).

CHEMISTRY 211(4), 212(4). Do *not* take 101/102 or 103/104.

ENVIRONMENTAL SCIENCE (EVPP) 110(4), 111(4).

GEOLOGY 101(4), 102(4).

PHYSICS 160(3)/161(1), 260(3)/261(1). You may substitute PHYS 265 for PHYS 261. This is "University Physics" and is strongly recommended. "College Physics" (243/244 and 245/246) can satisfy this requirement.

Computer Science Courses (CS)

One year of high school algebra is required for all courses; two years is strongly recommended.

101 Preview of Computer Science (2:2:0**)** *Co-requisite: CS 112.* All computer science majors are required to take this course within their first year. Offers a broad overview of computer science designed to provide students with an introduction to the field of computer science and an orientation to the computer science department and the computing environment at the university. Includes a project to introduce problem solving using computers.

105 Computer Ethics and Society (1:1:0). *Prerequisite: 12 credits of undergraduate coursework.* Intensive introduction to legal, social, and ethical issues surrounding software development and computer use. Stresses professional conduct, social responsibility, and rigorous standards for software testing and reliability. Examines issues such as liability, ownership of information, and computer crime.

112 Introduction to Computer Programming (4:3:1). *Prerequisite: Satisfaction of pre-requisites for MATH 113.* This course introduces the use of computer programming as a problem-solving tool. Topics in procedural programming include expressions, control structures, simple data types, input/output, graphical interfaces, testing, debugging and programming environments.

123 Computing: From the Abacus to the Web (3:3:0). *Prerequisites: none*. Students may not take this course for credit once they have successfully completed CS 211. This course, intended for non-majors, will give students the ability to relate to existing and emerging technologies (such as email, the Internet, search engines, weblogs, computer games and robotics) by educating them on the underlying computer science concepts. Historical, social and technical issues related to each topic will be discussed.

211 Object-Oriented Programming (3:3:1). *Prerequisite: CS 112.* This course continues to focus on problem solving, testing and debugging and introduces object-oriented programming. Topics include classes, inheritance, packages, collections, exceptions and polymorphism. Examples in the course may include the use of basic data structures.

222 Computer Programming for Engineers (3:3:0). *Prerequisites: CS 112.* A second course in computer programming. Introduces object-oriented programming, and elementary data structures. The emphasis will be on problems and language features relevant to engineers. (Intended as terminal course in computer programming.)

225 Culture and Theory of Games (3:3:0) Explores the theory, history, culture, and lore of games with particular emphasis on the varieties of computer game environments

261 Introduction to a Second Language (1:1:0). *Prerequisite: Grade of C or better in CS 211*. Advanced programming using Java programming language; other languages may be offered at times. *This course is not available for CS major credit.*

262 Introduction to Low-Level Programming (1:1:0) *Prerequisite: Grade of C or better in CS 211.* Introduction to the language C, as well as operating system concepts, in UNIX, to prepare students for topics in systems programming.

306 Synthesis of Ethics and Law for the Computing Professional (3:3:0). *Prerequisites: CS 105; junior standing (at least 60 credit hours). Co-requisite: all required General Education courses.* Computers science majors may use this course to satisfy the general education synthesis requirement, so long as they have not previously taken CS 305 for credit. Practical course to become effective computer professional. Examines legal, ethical issues surrounding computer technology and its use, as well as the foundation building that is necessary to deal with those challenges. Applies philosophical bases for ethical decision-making to modern concerns raised by computers and technology. Addresses topics covered by CS 105 in a more intensive manner, and focuses on the emerging legal and ethical issues involved in e-commerce and widespread use of the Internet.

310 Data Structures (3:3:0). *Prerequisite: CS 211. Co-requisite: CS 105.* This course continues to focus on object-oriented programming with an emphasis on tools and techniques for developing moderate to large programs. Topics include use and implementation of linear and non-linear data structures and the design and analysis of elementary algorithms.

325 Introduction to Game Design (3:3:0) *Prerequisite: CS 211.* Game design, in various electronic entertainment technologies, involves a diverse set of skills and backgrounds, from narrative and art to computer programming This course surveys the technical aspects of the field, with an emphasis on programming.

330 Formal Methods and Models (3:3:0). *Prerequisite: Grade of C or better in CS 211 and MATH 125.* Abstract concepts that underlie much advanced work in computer science, with major emphasis on formal languages, models of computation, logic, and proof strategies.

332 (CS/SWE) Object-Oriented Software Design and Implementation (3:3:0). *Prerequisite: CS 211.* In-depth study of software design and implementation using a modern, object-oriented language with support for graphical user interfaces and complex data structures. Topics covered will be specifications, design patterns, and abstraction techniques, including typing, access control, inheritance, and polymorphism. Students will learn to the proper engineering use of techniques such as information hiding, classes, objects, inheritance, exception handling, event-based systems, and concurrency.

363 Comparative Programming Languages (3:3:0). *Prerequisite: Grade of C or better in CS 367.* Key programming mechanics described independently of particular machines or languages including control, binding, procedural abstraction, and types. Systematically surveys diverse high-level language capabilities.

367 Computer Systems and Programming (3:3:0).). *Prerequisite: Grade of C or better in CS262 or 222 and ECE 301 or ECE 331.* Introduces students to computer systems from the perspective of a programmer. Topics covered include data representation, assembly and machine level representation of high-level language programs, the memory hierarchy, linking, exceptions, interrupts, processes and signals, virtual memory, and system-level I/O. This course serves as a foundation for courses on compilers, networks, operating systems, and computer architecture, where a deeper understanding of systems-level issues is required.

391 Advanced Programming Lab (1:0:1). *Co-requisite: Grade of C or better in CS 310 and permission of instructor*. Programming-intensive lab course. Students refine their problem solving and programming skills, while gaining experience in teamwork. Focuses on data structures, recursion, backtracking, dynamic programming, and debugging. Central focus is applying familiar and new algorithms and data structures to novel circumstances.

421 (CS/SWE) Software Requirements and Design Modeling (3:3:0). *Prerequisite: CS 211.* An introduction to concepts, methods, and tools for the creation of large-scale software systems. Methods, tools, notations, and validation techniques to analyze, specify, prototype, and maintain software requirements. Introduction to object-oriented requirements modeling, including use of case modeling, static modeling, and dynamic modeling using the Unified Modeling Language (UML) notation. Concepts and methods for the design of large-scale software systems. Fundamental design concepts and design notations are introduced. A study of object-oriented analysis and design modeling using the UML notation. Students participate in a group project on software requirements, specification, and object-oriented software design.

CS425 Game Programming I (3:3:0) *Prerequisites: CS 325 and 367.* The course will provide an introduction to technologies and techniques used in modern computer games. Teams will explore the various facets of a complete design, using sophisticated tools. The course will involve a project in which a game is prototyped; this prototype and initial design will serve as the starting point for the project in CS 426.

CS426 Game Programming II (3:3:0) *Prerequisite: CS 425.* This is a project-oriented course. It is a continuation of CS 425 with an emphasis on the implementation of a complete game.

440 Language Processors and Programming Environments (3:3:0). *Prerequisite: Grade of C or better in CS 310, 330 and 367*. Survey of basic programming language processors and software development tools, such as assemblers, interpreters, compilers. Topics include design and construction of language processors, formal syntactic definition methods, parsing techniques, and code generation techniques.

450 Data Base Concepts (3:3:0). *Prerequisite: Grade of C or better in CS 310 and 330.* This course covers from basics to intermediate knowledge for the design, implementation and use of relational database systems. The main topics include the Entity-Relationship (ER) and Entity-Enhanced Relationship (EER) models for database design, Relational Algebra (RA), Structured Query Language (SQL), SQL programming techniques, functional dependencies and normalization, object- and object-relational databases, and security. Students will practice to design, develop, and implement a relational ORACLE database, and use the database for queries, transaction processing, and report generation.

451 Computer Graphics (3:3:0). *Prerequisite: Grade of C or better in CS 310 and 367, and MATH 203.* Basic graphics principles and programming. Topics include scan conversion, transformation, viewing, lighting, blending, texture mapping, and some advanced graphics techniques.

455 Computer Communications and Networking Systems (3:3:0). *Prerequisite: Grade of C or better in CS 310 and 367, and STAT 344.* Data communication and networking protocols, with study organized to follow layers of the Internet Protocol Suite (TCP/IP family of protocols). Topics include role of various media and software components, local and wide area network protocols, network performance, and emerging advanced commercial technologies.

465 Computer Systems Architecture (3:3:0). *Prerequisite: grade of C or better in CS 367.* Computer subsystems and instruction set architectures. Single cycle, multiple-cycle, and pipeline architectures. Memory hierarchy, cache, and virtual memory input-output processing.

468 Secure Programming and Systems (3:3:0). *Prerequisite: grade of C or better in CS 310 and 367, or permission of the instructor.* Fundamental principles and techniques for implanting secure computer systems. Topics include security and cryptography basics, vulnerability analysis, secure software development and distributed system security. Projects involve designing and programming basic security tools, secure programs and distributed systems.

471 Operating Systems (3:3:0). *Prerequisites: Grade of C or better in CS 310 and 367*. Issues in multiprogramming. Covers concurrent processes and synchronization mechanisms; processor scheduling; memory, file, I/O, deadlock management; performance of operating systems. Projects dealing with synchronization in a multi-programmed OS and virtual memory management.

475 Concurrent and Distributed Systems (3:3:0). *Prerequisite: Grade of C or better in CS 310 and 367, or permission of instructor.* Practical issues in designing and implementing distributed software. Topics include concurrent programming, synchronization, multithreading, local and wide-area network protocols, distributed computation, system integration, and techniques for expressing coarse-grained parallelism at the application level. Projects involve network programming at the application level.

480 Introduction to Artificial Intelligence (3:3:0). *Prerequisite: Grade of C or better in CS 310 and 330.* Principles and methods for knowledge representation, reasoning, learning, problem solving, planning, heuristic search, natural language processing, and their application to building intelligent systems in a variety of domains. Uses LISP, PROLOG, or expert system programming languages.

482 Computer Vision (3:3:0). *Prerequisite: Grade of C or better in MATH 203, STAT 344 and CS 310.* Basic principles of visual perception and their implementation on computer systems. Topics include early visual processing, edge detection, segmentation, intrinsic images, image modeling, representation of visual knowledge, and image understanding. Students complete projects involving real images.

483 Analysis of Algorithms (3:3:0). *Prerequisite: Grade C or better in MATH 125 and CS 310 and 330.* Analyzes computational resources for important problem types by alternative algorithms and their associated data structures, using mathematically rigorous techniques. Specific algorithms are analyzed and improved.

484 Data Mining (3:3:0). *Prerequisite: Grade of C or better in CS 310 and STAT 344, or permission of the instructor.* Basic principles and methods for data analysis and knowledge discovery. Emphasizes developing the basic skills needed for modeling and prediction, on one side, and performance evaluation, on the other. Topics include system design; data quality, preprocessing, and association; event classification; clustering; biometrics; business intelligence; and mining complex types of data.

490 Design Exhibition (3:3:0). *Prerequisite: CS 421, 483, and two other CS 400-level courses and senior standing.* Capstone course focusing on the design and successful implementation of a major software project, encompassing a broad spectrum of knowledge and skills, developed by a team of students. Requires final exhibition to faculty-industry panel.

498 Independent Study in Computer Science (1-3:0:0). *Prerequisite: 60 credit hours, major in computer science, and permission of instructor.* Research and analysis of selected problems or topics in computer science. Topic must be arranged with and instructor and approved by the department chair before registering. May be repeated for a maximum of 6 credits if the topics are substantially different.

499 Special Topics in Computer Science (3:3:0). *Prerequisite: 60 credit hours and permission of instructor; specific prerequisites vary with nature of topic.* Topics of special interest to undergraduates. May be repeated for a maximum of 6 credits if the topics are substantially different.

BS/Accelerated MS Degree Programs

These programs are for students interested in immediately continuing on to graduate studies in computer science:

BS/Accelerated MS in Computer Science
BS/Accelerated MS in Information Security and Assurance
BS/Accelerated MS in Information Systems
BS/Accelerated MS in Software Engineering

Admission Requirements: Students in the BS program in computer science can apply for a BS/Accelerated MS program if they have earned 90 undergraduate credits with an overall GPA of at least 3.50 for the MS in Computer Science degree program or an overall GPA of at least 3.30 for the remaining MS degree programs. Criteria for admission are identical to criteria for admission to the respective MS program.

Degree Requirements: Students must complete 144 credits that satisfy requirements for the BS program as well as those for the MS program, with 6 credits overlap. Students register for 6 credits of CS 500-level basic courses in place of the corresponding CS 400-level courses required for the undergraduate degree requirements. That is, students must register for two of the following courses: CS 540, 571, 580, and 583 in place of the corresponding 400-level courses. Students complete all MS requirements as specified by the respective MS degree program and apply the two courses from the above list toward their MS requirements. Students are permitted to take additional graduate basic courses in their undergraduate programs. In such cases, those classes cannot be counted towards requirements for the MS.

Degree Conferral: Students must apply to have the BS degree conferred the semester before they expect to complete the BS requirements. At the completion of the MS requirements, a master's degree is granted.

Additional information about these and other accelerated degree options is available from the Computer Science Department office or at http://cs.gmu.edu.