COMPUTER SCIENCE

The Computer Science program provides a firm foundation in the theory and application of computing while allowing for additional concentration in areas of choice, such as information systems, mainframe computing, computer networking, telecommunications, data and knowledge engineering, and software development. This discipline is based on building software tools that make computers useful.

Mission and Vision
The mission of the Department is

• to provide outstanding undergraduate and graduate education in computer science;
• to conduct research that advances the state of the art in computer science, to integrate our educational, research, and service activities with other programs in the College of Information Science and Technology, the University of Nebraska at Omaha, and the communities we serve to reflect the role of computer science in information science and technology; and
• to ensure our efforts are of value and relevance to those we serve by continual assessment and improvement.

The vision of the Department is to be recognized nationally and internationally for delivering outstanding computer science education and conducting research of high distinction, both of value and relevance to the communities we serve.

Accreditation
The Computer Science program is accredited by the Computing Accreditation Commission of ABET, Inc., which is the recognized accreditor of college and university programs in applied science, computing, engineering, and engineering technology. ABET accreditation demonstrates a program’s commitment to providing its students with a quality education.

General Guidelines
Time of Admission to the Program
Students will be eligible for admission to the integrated degree program when they have completed their junior year in the College of IS&T; they can apply for consideration in the last part of their junior year. Students will start taking courses in the graduate program in their senior year.

Joint Admission
Students must apply to and meet admission requirements of the MS in CS graduate program.

Advising and Plan of Study
In consultation with an advisor and a faculty mentor, students will be required to prepare a plan of study. The plan will cover the entire time period of the program, and it will be periodically reviewed with an advisor. Students will present their portfolio and a plan of study in person to the IUG-CS Program Advisory Committee prior to being admitted to the program.

Tuition Charges
Students will be required to pay graduate tuition rates when taking graduate courses.

Admission Requirements and Procedures
1. Students with junior standing and at least 85 hours of completed course work in their undergraduate CS degree program may apply for admission consideration into the IUG-CS track.
   2. At the time of application, a student must have a GPA in their CS major area (or equivalent to the UNO CS major for transfer students) greater than or equal to 3.50 and an overall GPA of 3.50 or above.
   3. Interested students will be required to present a portfolio of the following credentials; whenever possible, this presentation will be made to the IUG-CS Program Advisory Committee:
      • Two letters of recommendations, at least one from faculty.
      • Statement of intent about why the student wishes to apply for the IUG-CS track.
      • Undergraduate transcripts.
      • GRE GENERAL score for non-UNO transfer students.
      • Other supporting documents (e.g., projects and papers, research activities, software, work experience, etc.) should be included where possible.
   4. Students are highly encouraged to identify and work with a faculty mentor who knows their background and can champion their application to the IUG-CS track.
   5. All applicants will need to meet any other admission requirements established for the MS in CS program.

Other Requirements
• The application to the IUG-CS track will be considered as a complete package, and therefore, obtaining a high undergraduate GPA and/or GRE GENERAL score is not a guarantee of admission.

Students are allowed to use a maximum of 6 hours of 8xx6 and 6 hours of 8xx0 courses towards the core extension requirements of the undergraduate degree, provided the courses are not the graduate equivalent of the CS core undergraduate requirements. The choice of graduate level courses in the core extension must also meet the requirements of the concentration plan for the integrated program.

Contact
For more information, contact the College of IS&T Academic Advising Office at 402-554-3819.

Website (http://www.unomaha.edu/college-of-information-science-and-technology/academics/computer-science.php)

Writing in the Discipline
All UNO students are required to take a writing in the discipline course within their major. Computer Science degree students must take CIST 3000.

Degrees Offered
• Computer Science, Bachelor of Science (http://catalog.unomaha.edu/undergraduate/college-information-science-technology/computer-science/computer-science-bs)

Bachelor of Science in Computer Science (BSCS)
The Bachelor of Science in Computer Science provides students with a solid background in the fundamentals of computing and prepares each individual for employment in a wide variety of positions and for graduate study in computer science. The content of the Department’s courses is continually monitored to ensure they are consistent with the fast-changing developments in the discipline. Courses are offered in day and evening sections for the convenience of the students. Appropriate university and departmental computing resources are available to students taking computer sciences courses.

A minimum of 120 credit hours is required for a Bachelor of Science degree in Computer Science (BSCS). Thirty of the last 36 hours must be University of Nebraska at Omaha courses. Registering for courses without having taken the stated prerequisites could result in administrative withdrawal.
Second Baccalaureate Degree

General Requirements

Students who have satisfied the requirements for a first baccalaureate degree other than computer science at the University of Nebraska at Omaha must complete a minimum of 30 additional semester hours at the University for a second bachelor’s degree.

Second Degree Requirements

To obtain computer science as a second bachelor’s degree, students must complete academic requirements for the degree which include 18 credit hours of IS&T core courses, 27 credit hours of computer science core courses, 21 credit hours of a computer science core extension, and 16 credit hours of Mathematics courses, provided that the first baccalaureate degree is not in computer science. Students who are admitted to a second degree program in CS must meet with an academic advisor in the College of IS&T before beginning the degree to make a plan of study. Some transfer coursework may apply; however, 30 of the last 36 hours must be University of Nebraska at Omaha courses.

Minors Offered

- Computer Science Minor (http://catalog.unomaha.edu/undergraduate/college-information-science-technology/computer-science/computer-science-minor)

Computer Science Minor

A minor in computer science may be earned by completing 12 hours of the IS&T core courses (CIST 1400, CSCI 1620, CIST 2100 and CIST 2500), including 6 hours of computer science courses at the 2000 level or above and 6 hours at the 3000 level or above.

Students are accountable for prerequisite courses.

Other Information

Integrated Undergraduate/Graduate Track (IUG) in Computer Science

The Department of Computer Science offers an integrated undergraduate/graduate track that provides undergraduate students in the College of IS&T the option to complete the undergraduate degree (BS) in computer science (CS) and the graduate degree (MS) in computer science in five years. The program is a 141 to 144 hour undergraduate-graduate program that allows eligible students who are motivated and willing to take on early the challenges related to graduate education to work toward an accelerated MS degree in CS while completing their undergraduate degree. Students interested in this program will work closely with an adviser and a faculty mentor to develop an integrated plan of study. The program also allows students to follow either a general computer science study plan or a specialized concentration study plan. Seven areas of specialized concentration are provided in the graduate CS program:

1. Artificial Intelligence
2. Databases and Knowledge Engineering
3. Programming Languages
4. Network Technologies
5. Software Engineering
6. Information Systems
7. Information Assurance

The area of concentration will be noted on the student’s transcript.

The program offers three degree options:

1. Thesis
2. Project
3. Coursework

CSCI 1200 COMPUTER SCIENCE PRINCIPLES (3 credits)
This course introduces students to the foundational principles of computer science. It aims to help students learn the essential thought processes used by computer scientists to solve problems, expressing those solutions as computer programs. The exercises and projects make use of mobile devices and other emerging platforms.
Prerequisite(s)/Corequisite(s): MATH 1310 or equivalent.
Distribution: Natural/Physical Sci General Education lecture

CSCI 1204 COMPUTER SCIENCE PRINCIPLES LABORATORY (1 credit)
This is a laboratory course for students enrolled in CSCI 1200. It consists of programming exercises designed to help students practice computational thinking and apply computational solutions to practical problems. The exercises make use of mobile devices and other emerging platforms.
Prerequisite(s)/Corequisite(s): CSCI 1200, prior or concurrent.
Distribution: Natural/Physical Sci General Education lab course

CSCI 1620 INTRODUCTION TO COMPUTER SCIENCE II (3 credits)
Advanced topics in programming; topics in data representation and manipulation, data structures, problem solving and algorithm design. This course has a required laboratory component; students must register for a laboratory section when enrolling in lecture.
Prerequisite(s)/Corequisite(s): CIST 1400 and MATH 1930 or MATH 1950 (with a grade of "C" or better)

CSCI 2030 MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE (3 credits)
This course introduces discrete mathematics concepts that are foundational for the study of computer science such as functions, relations, and sets, basic logic, methods of proof, mathematical induction, computational complexity, recursion, counting, recurrences, and relations.
Prerequisite(s)/Corequisite(s): CIST 1400, MATH 1950 or MATH 1930

CSCI 2240 INTRODUCTION TO C PROGRAMMING (3 credits)
Programming in 'C' in a UNIX operating system environment; algorithm and program development and file manipulation using 'C'; UNIX-like utility development.
Prerequisite(s)/Corequisite(s): CSCI 1620; Facility w/high-level prog lang like Pascal, Modula, Java, or C++; Ability to design & implement solutions to modest problems.

CSCI 2310 VIDEO GAME DESIGN (3 credits)
The course will cover game design and theory techniques used by the gaming industry. This course is designed for students who have gone through the introductory programming course and have an interest in what it takes to design current games.
Prerequisite(s)/Corequisite(s): CIST 1400.

CSCI 2510 INTRODUCTION TO GAME PROGRAMMING (3 credits)
The course will cover programming and development techniques used in a game programming environment. The course is designed for students who have an interest in game programming to be eased into the concepts in a familiar environment.
Prerequisite(s)/Corequisite(s): CSCI 2240. Not open to non-degree graduate students.

CSCI 2840 C++ & OBJECT-ORIENTED PROGRAMMING (3 credits)
C++ and Object Oriented Programming (OOP) is taught in the UNIX environment. Topics include C++ as a 'Better C,' OOP with C++, classes and data abstraction, operator overloading, inheritance, virtual functions and polymorphism, C++ stream I/O.
Prerequisite(s)/Corequisite(s): CSCI 2240; High-level programming language like Pascal, Java, or C++; solid understanding of pointers & scope; ability to design & implement solutions to modest problems.
CSCI 2850 PROGRAMMING ON THE INTERNET (3 credits)
This course is an introduction to and overview of Internet-based application
development focusing on the use of Java, Perl and other server-based
programming languages. Software development in the context of the
World Wide Web and other Internet services will be emphasized. Internet
application development will also be discussed. Other techniques will be
covered.
Prerequisite(s)/Corequisite(s): CSCI 1620 or CSCI 1840.

CSCI 2980 TOPICS IN COMPUTER SCIENCE (1-3 credits)
A variable topic course in computer science at the sophomore level. Topics
not covered in the computer science degree program, but suitable for
sophomore-level students.
Prerequisite(s)/Corequisite(s): Permission of instructor. Additional
prerequisites may be required for particular topic offerings.

CSCI 3100 APPLIED COMBINATORICS (3 credits)
Basic counting methods, generating functions, recurrence relations,
principle of inclusion-exclusion, Polya’s formula. Elements of graph theory,
trees and searching network algorithms. (Cross-listed with CSCI 8105,
MATH 3100, MATH 8105).
Prerequisite(s)/Corequisite(s): MATH 2030 with a C- or better
or MATH 2040 with a C- or better or MATH 2230 with a C- or better.
Mathematical logic; Set theory; Relations; Functions; Congruences;
Inductive and recursive definitions; Discrete probability; sets, graphs, trees,
& matrices

CSCI 3300 NUMERICAL METHODS (3 credits)
This course involves solving nonlinear algebraic equations and systems
of equations, interpolation and polynomial approximation, numerical
differentiation and integration, numerical solutions to ordinary differential
equations, analysis of algorithms and errors, and computational efficiency.
(Cross-listed with CSCI 8305, MATH 3300, MATH 8305).
Prerequisite(s)/Corequisite(s): MATH 1960 with a C- or better
or permission of instructor

CSCI 3320 DATA STRUCTURES (3 credits)
This is a core course that will cover a number of data structures such as tree,
hashing, priority queues and graphs as well as different algorithm design
methods by examining common problem-solving techniques. (Cross-listed
with CSCI 8325)
Prerequisite(s)/Corequisite(s): CSCI 1620 and CSCI 2030 or
MATH 2030. Programming Languages: Java or C++ Topics: Arrays, Pointers,
Introductory Lists, Storage Allocation

CSCI 3450 NATURAL LANGUAGE PROCESSING (3 credits)
The course will provide overview of the topics in natural language
processing such as word and sentence tokenization, syntactic parsing,
semantic role labeling, text classification. We will discuss fundamental
algorithms and mathematical models for processing natural language, and
how these can be used to solve practical problems. We will touch on such
applications of natural language processing technology as information
extraction and sentiment analysis. (Cross-listed with CYBR 3450)
Prerequisite(s)/Corequisite(s): Prerequisite: CSCI 2030; Co-requisite:
CSCI 3320; Students should be comfortable w/ scripting (Python will be the
language extensively used in natural language processing tools including
NLTK). Not open to non-degree graduate students.

CSCI 3510 ADVANCED GAME PROGRAMMING (3 credits)
This course is intended for those with an interest in video game
programming. This course introduces the advanced concepts of game
programming including 3D programming, game networking, and
development of a multiplayer, networked game by learning and using the
XNA environment.
Prerequisite(s)/Corequisite(s): CSCI 2510 and CSCI 3320 or Instructor
permission. Not open to non-degree graduate students.

CSCI 3550 COMMUNICATION NETWORKS (3 credits)
This course is designed to bring students up to the state of the art in
networking technologies with a focus on Internet. It will cover the principles
of networking with an emphasis on protocols, implementations and design
issues. (Cross-listed with CSCI 8555)
Prerequisite(s)/Corequisite(s): CSCI 3320 or CSCI 8325. Data structures
and algorithms. C or C++ programming.

CSCI 3660 THEORY OF COMPUTATION (3 credits)
The course is intended to introduce the students to the theory of
computation in a fashion that emphasizes breadth and away from detailed
analysis found in a normal undergraduate automata course. The topics
covered in the course include methods of proofs, finite automata, non-
determinism, regular expressions, context-free grammars, pushdown
automata, no-context free languages, Church-Turing Thesis, decidability,
reducibility, and space and time complexity.
Prerequisite(s)/Corequisite(s): CSCI 3320

CSCI 3710 INTRODUCTION TO DIGITAL DESIGN AND COMPUTER
ORGANIZATION (3 credits)
The course is intended to introduce the students to the topics found in
introductory digital design and computer organization classes.
Prerequisite(s)/Corequisite(s): CSCI 3320 (could be taken concurrently)

CSCI 3830 ADVANCED JAVA PROGRAMMING (3 credits)
This course teaches students Web-based programming techniques in the
Java programming language. It begins with programming using traditional
models such as the client-server model and then transitions to advanced
programming paradigms including middleware programming using RMI
and CORBA, the distributed JINI/JavaSpaces model, the peer-to-peer
networking model and the agent-based programming model.
Prerequisite(s)/Corequisite(s): CSCI 1620

CSCI 3850 FOUNDATIONS OF WEB SEARCH TECHNOLOGIES (3
credits)
This course provides students a basic understanding of how search
and information flow works on the web. Main topics include: document
representation, inverted indexing, ranking of web search results, vector-
space model, web graph, PageRank, search-based advertising, information
cascades, and web crawling.
Prerequisite(s)/Corequisite(s): CSCI 2030 and CSCI 2850, or instructor
Approval. Not open to non-degree graduate students.

CSCI 4000 ASSESSMENT (0 credits)
This course provides various resources to students about to graduate, and
provides a mechanism that guarantees these students complete the final
assessments required to maintain the currency and quality of the program.
It is intended for undergraduate computer science majors in their last
semester prior to graduation. It is required for all students entering after
the spring 2004 semester. All degree requirements should be completed by
the end of the semester during which this course is taken. Students taking
this course will be expected to file an application for graduation during the
semester.
Prerequisite(s)/Corequisite(s): All degree requirements should be
completed by the end of the semester during which this course is taken.
Students taking this course will be expected to file an application for
graduation during the semester.

CSCI 4010 INTRODUCTION TO THE THEORY OF RECURSIVE
FUNCTIONS (3 credits)
This is a proof-oriented course presenting the foundations of Recursion
Theory. We present the definition and properties of the class of primitive
recursive functions, study the formal models of computation, and
investigate partially computable functions, universal programs. We
prove Rice’s Theorem, the Recursion Theorem, develop the arithmetic
hierarchy, demonstrate Post’s theorem. Introduction to the formal theories
of computability and complexity is also given. (Cross-listed with MATH 4010,
MATH 8016, CSCI 8016).
Prerequisite(s)/Corequisite(s): CSCI 2230 or CSCI 3660 or instructor’s
permission
CSCI 4100 INTRODUCTION TO ALGORITHMS (3 credits)
The course provides students a basic understanding of algorithm analyses. Main topics include: growth of functions, asymptotic notation, recurrences, divide and conquer, sorting and its lower bounds, dynamic programming, greedy algorithms, and graph traversal.
Prerequisite(s)/Corequisite(s): CSCI 3320.

CSCI 4150 GRAPH THEORY & APPLICATIONS (3 credits)
Introduction to graph theory. Representations of graphs and graph isomorphism. Trees as a special case of graphs. Connectivity, covering, matching and coloring in graphs. Directed graphs and planar graphs. Applications of graph theory in several fields such as networks, social sciences, VLSI, chemistry and parallel processing. (Cross-listed with CSCI 8156, MATH 4150, MATH 8156).
Prerequisite(s)/Corequisite(s): MATH 2030 or permission of instructor.

CSCI 4200 PRINCIPLES OF PROGRAMMING LANGUAGES (3 credits)
This course covers the foundational concepts and principles underlying the design and implementation of programming languages. Language constructs including assignment, equality, references, aggregations, scope, encapsulation, and parameter passing are discussed. A central theme is how a particular language construct relates to the concept of equational reasoning (referential transparency). Formal notations for describing syntax and semantics are presented.
Prerequisite(s)/Corequisite(s): CSCI 3320, CSCI 3660.

CSCI 4250 HUMAN COMPUTER INTERACTION (3 credits)
Human computer interaction is concerned with the joint performance of tasks by humans and machines; human capabilities to use machines (including learnability of interfaces); algorithms and programming of the interface; engineering concerns that arise in designing and building interfaces; the process of specification, design, and implementation of interfaces; and design trade-offs. (Cross-listed with CSCI 8256).
Prerequisite(s)/Corequisite(s): CSCI 4830 (may be taken concurrently).
Knowledge of object-oriented programming concepts. Demonstrated fluency in any visual programming language.

CSCI 4260 USER INTERFACE DESIGN AND DEVELOPMENT (3 credits)
Graphical user interface (GUI) design is concerned with the application of user-centered design principles to graphical computer interfaces. Topics include user-centered design, establishing usability criteria and measures, usability testing, psychology of the user, rapid prototyping, iterative design and design tools. This course is an extension and application of its prerequisite, Human Computer Interaction. (Cross-listed with CSCI 8266).
Prerequisite(s)/Corequisite(s): CSCI 4250 or instructor’s permission.
CSCI 4260 Knowledge of object-oriented programming concepts, Visual programming language.

CSCI 4300 DETERMINISTIC OPERATIONS RESEARCH MODELS (3 credits)
This is a survey course of deterministic operations research models and algorithms. Topics include linear programming, network programming, and integer programming. (Cross-listed with CSCI 8306, MATH 4300, MATH 8306).
Prerequisite(s)/Corequisite(s): MATH 2050 with a C- or better or permission of instructor.

CSCI 4310 PROBABILISTIC OPERATIONS RESEARCH MODELS (3 credits)
This is a survey course of probabilistic operations, research models and algorithms. Topics include Markov chains, queueing theory, inventory models, forecasting, and simulation. (Cross-listed with CSCI 8316, MATH 4310, MATH 8316).
Prerequisite(s)/Corequisite(s): MATH 2050 and either MATH 4740 or MATH 8746 or STAT 3800 or STAT 8805 all with a C- or better or permission of instructor.

CSCI 4350 COMPUTER ARCHITECTURE (3 credits)
The course deals with: processor design different instruction set architectures; memory hierarchy; input output organization and communication; and an introduction to parallel architecture. Analytic study of design alternatives for each of the above topics will be covered. The course is designed primarily for third year students in Computer Science.
Prerequisite(s)/Corequisite(s): CSCI 3710, CSCI 3320 or CSCI 8325.

CSCI 4380 COMPUTER AND NETWORK FORENSICS (3 credits)
Computer forensics involves the preservation, identification, extraction and documentation of computer evidence stored on a computer. This course takes a technical, legal, and practical approach to the study and practice of incident response, computer forensics, and network forensics. Topics include legal and ethical implications, duplication and data recovery, steganography, network forensics, and tools and techniques for investigating computer intrusions. This course is intended as a second course in information assurance for undergraduate students as well as other qualified students. It is also intended as a foundation course for graduate digital forensics studies. (Cross-listed with CYBR 4380, CYBR 8386).
Prerequisite(s)/Corequisite(s): CYBR 1100, CIST 3600, CSCI 3550 or ISQA 3400, CYBR 3350 or CYBR 3370; or instructor permission.

CSCI 4440 INTRODUCTION TO PARALLEL COMPUTING (3 credits)
Need for higher-performance computers. Topics discussed include: classification of parallel computers; shared-memory versus message passing matchings; for ms of parallelism, measure of performance; designing parallel algorithms; parallel programming and parallel languages; synchronization constructs; and operating systems for parallel computers. (Cross-listed with CSCI 8446).
Prerequisite(s)/Corequisite(s): CSCI 4500 which may be taken concurrently.

CSCI 4450 INTRODUCTION TO ARTIFICIAL INTELLIGENCE (3 credits)
An introduction to artificial intelligence. The course will cover topics such as machine problem solving, uninformed and informed searching, propositional logic, first order logic, approximate reasoning using Bayesian networks, temporal reasoning, planning under uncertainty and machine learning. (Cross-listed with CSCI 8456).
Prerequisite(s)/Corequisite(s): CSCI 3320.

CSCI 4470 PATTERN RECOGNITION (3 credits)
Prerequisite(s)/Corequisite(s): CSCI 1620, and MATH 2050. Recommended: MATH 4740/8746 or STAT 3800/8805.

CSCI 4480 ALGORITHMS FOR ROBOTICS (3 credits)
This course provides an introduction to software techniques and algorithms for autonomously controlling robots using software programs called controllers. Students will be taught how to program and use software controllers on simulated as well as physical robots. (Cross-listed with CSCI 8486).
Prerequisite(s)/Corequisite(s): CSCI 4500 and MATH 4350/8350.

CSCI 4500 OPERATING SYSTEMS (3 credits)
Operating system principles. The operating system as a resource manager; I/O programming, interrupt programming and machine architecture as it relates to resource management; memory management techniques for uni-multiprogrammed systems; process description and implementation; processor management (scheduling); I/O device, controller, and channel management; file systems. Operating system implementation for large and small machines. (Cross-listed with CSCI 8506).
Prerequisite(s)/Corequisite(s): CSCI 3710, CSCI 3320/8325, MATH 1950, and CSCI 4350/8350.
CSCI 4510 ADVANCED OPERATING SYSTEMS (3 credits)
State-of-the-art techniques for operating system structuring and implementation. Special purpose operating systems. Pragmatic aspects of operating system design, implementation and use. (Cross-listed with CSCI 8516)
Prerequisite(s)/Corequisite(s): CSCI 4500

CSCI 4560 NUMBER THEORY & CRYPTOGRAPHY (3 credits)
An overview of one of the many beautiful areas of mathematics and its modern application to secure communication. The course is ideal for any student who wants a taste of mathematics outside of, or in addition to, the calculus sequence. Topics to be covered include: prime numbers, congruences, perfect numbers, primitive roots, quadratic reciprocity, sums of squares, and Diophantine equations. Applications include error-correcting codes, symmetric and public key cryptography, secret sharing, and zero knowledge proofs. (Cross-listed with CSCI 8566, MATH 4560, MATH 8566).
Prerequisite(s)/Corequisite(s): MATH 2230 with a C- or better or MATH 2030 with a C- or better or MATH 2030 with a C- or better or permission of instructor

CSCI 4620 COMPUTER GRAPHICS (3 credits)
An introduction to the acquisition, manipulation and display of graphical information using digital techniques. Topics include discussion of the various hardware devices used for input and output, the classical algorithms and data structures used in manipulation of graphical objects, the user interface to the graphics system, and applicable standards. (Cross-listed with CSCI 8626).
Prerequisite(s)/Corequisite(s): ISQA 3300 or CSCI 3320

CSCI 4660 AUTOMATA, COMPUTABILITY, AND FORMAL LANGUAGES (3 credits)
This course presents a sampling of several important areas of theoretical computer science. Definition of formal models of computation, and important properties of such models, including finite automata and Turing machines. Definition and important properties of formal grammars and their languages. Introduction to the formal theories of computability and complexity. (Cross-listed with CSCI 8666, MATH 4660, MATH 8666).
Prerequisite(s)/Corequisite(s): MATH 2030. Recommended: CSCI 3320/ CSCI 8325.

CSCI 4700 COMPILER CONSTRUCTION (3 credits)
Assemblers, interpreters and compilers. Compilation of simple expressions and statements. Analysis of regular expressions. Organization of a compiler, including compile-time and run-time symbol tables, lexical scan, syntax scan, object code generation and error diagnostics. (Cross-listed with CSCI 8706).
Prerequisite(s)/Corequisite(s): CSCI 3320 and CSCI 4220. Recommended: CSCI 4500.

CSCI 4760 TOPICS IN MODELING (3 credits)
Selection of such topics as formulation and analysis of various models involving Markov chains, Markov processes (including birth and death processes), queues, cellular automata, difference and differential equations, chaotic systems and fractal geometries. (Cross-listed with CSCI 8766, MATH 4760, MATH 8766).
Prerequisite(s)/Corequisite(s): MATH 2350 and MATH 4740 or MATH 8746.

CSCI 4830 INTRODUCTION SOFTWARE ENGINEERING (3 credits)
Basic concepts and major issues of software engineering, current tools and techniques providing a basis for analyzing, designing, developing, maintaining and evaluating the system. Technical, administrative and operating issues. Privacy, security and legal issues. (Cross-listed with CSCI 8836).
Prerequisite(s)/Corequisite(s): CSCI 3320.

CSCI 4850 DATABASE MANAGEMENT SYSTEMS (3 credits)
Basic concepts of data base management systems (DBMSs). The relational, hierarchical and network models and DBMSs which use them. Introduction to data base design. (Cross-listed with CSCI 8856).
Prerequisite(s)/Corequisite(s): CSCI 3320

CSCI 4890 DATA WAREHOUSING AND DATA MINING (3 credits)
This course provides students with a theoretical foundation and practical methods for designing and constructing data warehouse and implementing data mining. After covering the essential concepts, issues, techniques to build an effective data warehouse, this course emphasizes the various techniques of data mining, such as association, classification, clustering and prediction for on-line analyses within the framework of data warehouse architectures. This course gives students an opportunity to undertake a real-life data analysis project. (Cross-listed with ISQA 4890).
Prerequisite(s)/Corequisite(s): ISQA 3310 or CSCI 4850

CSCI 4900 INTERNET SYSTEMS DEVELOPMENT (3 credits)
This course focuses on contemporary techniques and technologies in the design, development, and integration of web-enabled information systems. This is a rapidly moving, hands-on course that mirrors real-world development of internet-based applications.
Prerequisite(s)/Corequisite(s): CSCI 1620, CSCI 2850, (recommended) CSCI 3830, CSCI 4830,

CSCI 4950 INTERNSHIP IN COMPUTER SCIENCE (1-3 credits)
The purpose of this course is to provide students with opportunities to apply their academic studies in non-academic environments such as those found in business, industry and other non-academic organizations. The student intern will sharpen their academic focus and develop better understanding of non-academic application areas. The course is intended primarily for juniors and seniors in computer science.
Prerequisite(s)/Corequisite(s): Permission of the computer science program chair.

CSCI 4970 CAPSTONE PROJECT (3 credits)
The Capstone Project completes a Computer Science student’s undergraduate experience. Students will work on a team-based real-world project, practicing software engineering skills and applying fundamental computer science principles acquired throughout their undergraduate study.
Prerequisite(s)/Corequisite(s): CSCI 4830 ; Senior standing in Computer Science. Not open to non-degree graduate students.

CSCI 4980 TOPICS IN COMPUTER SCIENCE (1-3 credits)
A variable topic course in computer science at the senior level. Topics not normally covered in the computer science degree program, but suitable for senior-level students. (Cross-listed with CSCI 8986).
Prerequisite(s)/Corequisite(s): Permission of instructor. Additional prerequisites may be required for particular topic offerings.

CSCI 4990 INDEPENDENT STUDIES (1-3 credits)
A variable credit course for the junior or senior who will benefit from independent reading assignments and research type problems. Independent study makes available courses of study not available in scheduled course offerings. The student wishing to take an independent study course should find a faculty member willing to supervise the course and then submit, for approval, a written proposal (including amount of credit) to the Computer Science Undergraduate Program Committee at least three weeks prior to registration.
Prerequisite(s)/Corequisite(s): Written permission required.