COMPUTER SCIENCE PROGRAM

Program Director: Stuart A. Steele

Computer science is the study of the theory and practice of how to design, build and use computers. The field of study includes the design and analysis of algorithms, principles of programming languages and compilers, operating systems, software engineering, artificial intelligence, computer organization and architecture, computational geometry, database systems, parallel and distributed computing, and image analysis and understanding. The Computer Science Program is administered by the Department of Computer and Information Science.

UNDERGRADUATE PROGRAM

This program in computer science is accredited by the Computing Accreditation Commission (CAC) of the Accreditation Board for Engineering and Technology (ABET). The program in computer science offers a curriculum that prepares students for a professional career as computer scientists or graduate studies in computer science leading to research or teaching careers. The undergraduate program in computer science at Polytechnic is based on the principle that a well-rounded graduate should have a rigorous education in the fundamentals of computer science, including significant exposure to the design and operation of computers. Extensive hands-on experience with projects and teamwork, breadth and flexibility in elective courses, solid training in mathematics and science, and a general education based upon multidisciplinary courses are an integral part of the curriculum.

GOALS AND OBJECTIVES

The goals and objectives of the Bachelor of Science program in Computer Science are to provide students with the following:

- Strong fundamental science and mathematical base of knowledge
- Skills and fundamentals of computer science
- Proper balance of theory and practice for problem solving
- Well-balanced education with knowledge in humanities, mathematics, science, business and computer science
- Opportunity to explore various aspects of computer science and other fields through flexible electives

OVERVIEW OF THE UNDERGRADUATE CURRICULUM

The main focus of the undergraduate program is on theory, practice, interdisciplinary activities and problem-solving techniques. The curriculum addresses (1) the core/common knowledge that should be required of all computer science graduates and (2) a set of other important topics in computer science that students can choose to study depending on individual interests and career goals. For the required computer science core, courses are chosen in the following six areas: theoretical foundations of computer science, algorithms, data structures, software design, the concepts of programming languages and computer elements and architecture. For the computer science elective courses, the program offers a wide range of advanced topics, including java programming, UNIX programming, parallel and distributed processing, database systems design and interactive computer graphics.

The Computer Science Program allows students to receive a balanced education in the three processes of computing discipline: theory, abstraction and design/skill. Theory provides the underlying mathematical or scientific principles that apply to the discipline of computing. In the abstraction process, students develop models for potential algorithms, data structures, architectures and so forth. In the design process, students engage in the development of a computer system or software using necessary computer skills (e.g., proficiency in a particular programming language or database package).

Undergraduates in computer science at Polytechnic have the advantage of being in a department with a strong graduate division. This means that the undergraduate students study in a rich intellectual environment where many of their instructors are engaged in state-ofthe-art research. This significantly contributes to the quality of education and provides highly motivated undergraduates with the opportunity to engage in advanced projects with first-rate researchers.

COMPUTER SCIENCE COMPONENT (48 credits)

One of the distinctive features of the computer science component is the balance of emphasis on subjects related to the design of computers and theoretical computer science. For example, students study computer organization and architecture, as well as data structures, software development, database systems, operating systems, C++ and object oriented technologies, advanced algorithms and the principles of programming languages. The department believes that this balance of emphasis is important in preparing graduates for a professional or research career.

To work with a variety of students with different levels of computer experience, the Department of Computer and Information Science offers two different levels of CS 1114 Introduction

to Programming and Problem Solving: inexperienced and experienced. The experienced level is designed for students who already know some of the basics, such as keyboards, simple prompts, input/output, variables and simple loops, and who therefore would be bored if these were covered slowly. The course for inexperienced users will be paced for first-time users and will go more slowly through the basic material.

The elective strategy for the proposed BS Computer Science Program consists of eight electives: a two course sequence selected by students from the list of Elective Sequences designated by the University, four CS electives, one technical elective and one free elective.

At least two of the four required CS electives must be chosen from the following list:

•Artificial Intelligence I
•Computer Networking
•Computer Security
•Digital Logic and State Machine Design
•Introduction to Databases
•Introduction to Parallel and Distributed Systems

Computer science is an ever-changing field; the department regularly offers selected topics courses in current areas of computer technology; selected topics are announced every semester.

Mathematics Component (20 credits)

Mathematics is essential to the computer science curriculum. It forms the basis for understanding computer architecture and organization, principles of programming languages, algorithms, compilers and operating systems. The mathematics sequence is designed to enhance the integration of mathematics with the computer science component. If students did not have a chance to learn high school math well (as determined by the Polytechnic placement examination in mathematics), they will be placed in preparatory mathematics courses in order to prepare them for the calculus sequence. The physics sequence begins in the first term of the sophomore year to take advantage of students' preparation in mathematics.

Basic Science Component (12 credits)

Basic courses in physics and chemistry provide a well-rounded education in science. Computer scientists find that their training in basic science plays an important role in their career by allowing them to understand the theoretical principles of new devices.

Basic Engineering Component (4 credits)

Today, computers are used in all disciplines of engineering. Applications range from computer simulation of wind tunnels to computer-aided design (CAD) of automobile parts and loadflow analysis of electric power circuits. It is important that computer scientists can communicate with engineers from different disciplines to understand their needs. EG 1004 Introduction to Engineering and Design introduces computer science students to practical design experience in various disciplines of engineering.

Humanities and Social Sciences Component (30 credits)

Courses in the humanities and social sciences are an important part of the curriculum. Career advancement is based not only on technical skills and knowledge; it is equally based on the ability to communicate effectively and the ability to apply the wisdom that results from a serious study and appreciation of the humanities and social sciences. Thus, in addition to basic humanities and social sciences courses, the department requires students to take PL 4062 Computer Ethics.

University Elective Sequence (8 credits)

A list of approved Elective Sequences for computer science majors is available in the Department of Computer and Information Science.

Technical Elective and Free Elective Components (3 credits each)

It is important for students to have the opportunity to explore other subject areas or to delve into areas in more depth. This is the purpose of having 3 credits of technical electives and 3 credits of free electives.

MINOR IN COMPUTER SCIENCE

The minor in Computer Science consists of 15 credits. Students would need to obtain a grade of C- or better in CS 1114 and maintain an average of 2.0 GPA or better in the entire minor. Transfer students must take at least two of the four courses at Polytechnic University.

HONORS PROGRAM

Full-time students may apply for the BS/MS Accelerated Honors Program, which leads to the simultaneous award of a bachelor's and a master's degree. Depending on the student's preparation and objectives, completion of the two degrees may come as early as the end of three and three-quarter calendar years of study, or as late as five and one-half years. But each program is individually designed in cooperation with a departmental BS/MS Accelerated Honors Program adviser to allow for varied transfer and AP credits, co-op program participation, professional summer jobs and

other goals consistent with an honors program. Possible BS/MS combinations: a BS in Computer Science with an MS in Computer Science; a BS in Computer Engineering with an MS in Computer Science; and a BS in Electrical Engineering with an MS in Computer Science.

In order to be admitted to the BS/MS Honors Program, students must have exemplary academic achievements in high school, such as high GPAs, strong SAT scores and Advanced Placement credit in calculus and computer science. Students are invited to join the program by the department prior to the start of the first semester of their freshman year. If students are not accepted at that time, their application may be reconsidered after their first semester at Polytechnic by reapplying with the departmental Honors Program Director. Program participants must complete 16 to 20 credits each semester, maintain a technical and overall 3.5 GPA and display a record essentially free of course repetitions and withdrawals.

The required courses for the two degrees include all courses required for the individual BS and MS degrees, and all curriculum footnotes apply. Required credits are the sum of the credits for the two degrees, except that 3 free elective credits are excused. Six credits of Master's Thesis are required, and a special nine-week full-time summer honors research project at the end of the second or third year is urged, if offered.

Acceleration may be achieved through summer course work; research participation; extra course loads; careful course sequencing; and AP credit in such courses as MA 1024 and 1112 (AP Calculus BC, grade of at least 4 or 5); and CS 1114 (AP Computer Science A or AB, grade of 4 or 5).

TRANSFER STUDENTS

Transfer students are accepted into the Undergraduate Computer Science Program on the same basis described in the admissions section of this catalog. In addition, the department requires that at least 28 credits in computer science, as well as CS 4513 and CS 4523, be completed at Polytechnic. Graduates of technology programs may be able to fulfill the requirements for the BS in Computer Science in two to three-and-one-half years, depending on the scope and level of their previous education. Consult an undergraduate adviser for details.

Courses taken at other schools may be granted transfer credit based on evaluation of the content and level of material covered. Periodic reevaluation of courses at other institutions may lead to a variation in the amount of credits granted from year to year. Thus, students completing the same program, but in different years, may receive different amounts of transfer credit. Consult a computer science undergraduate adviser for current information. All computer science courses will be evaluated by the Department of Computer and Information Science. Transfer students who are exempted from EG 1004 Introduction to Engineering and Design must take a substitute course that includes presentation preparation. Students should meet with their undergraduate adviser for more information.

DEPARTMENTAL STANDARDS, PROBATIONS AND GRADES OF I (INCOMPLETE)

Computer scientists are professionals who are expected to achieve work of acceptable quality and quantity within a specified time. Similarly, Polytechnic students need to assure timely academic progress. It is this ability, the ability to work and to achieve, which is most desired by prospective employers.

To remain in good standing, computer science majors must satisfy the requirements listed below. These requirements are in addition to the University requirements for a minimum term and cumulative 2.0 GPA in all courses.

The following requirements apply to all undergraduate computer science students:

1. Students must maintain an average of C (2.0 GPA) or better in CS

2. Agrade of C- or better in the following courses: MA 1024 Calculus I; MA 1124 Calculus II; CS 1114 Introduction to Programming and Problem Solving; CS 1124 Object Oriented Programming; and CS 2134 Data Structures and Algorithms

3. Students may repeat a course in which they earned a substandard grade, but no CS course may be taken more than three times (grades of W and AUDIT are not counted for the purpose of this rule)

4. A course receiving an I grade may not be used to satisfy any prerequisites until the incomplete is resolved. See "Policies on Grading and Grades for the University" in this catalog for additional information on incomplete grades.

Students failing to meet any of the above requirements are placed on departmental probation as a warning that they are not progressing acceptably toward their degree. Repeated failure to meet probation requirements may lead to disqualification from the undergraduate computer science program and courses. If students have any questions, they should feel free to discuss them with an adviser, and preferably in a timely fashion so that good solutions can be found to any problems that may arise.

INFORMATION

Curricula and prerequisite changes, new courses, special sections and other last minute announcements are posted on the bulletin boards outside the offices of the Department of Computer and Information Science. Each stu-

dent is responsible for keeping informed of such changes.

List of CS Electives:

Undergraduate Courses

CS 205	Assembly Language
CS 239	Advanced UNIX System Programming
CS 308	Introduction to Databases
CS 342	Algorithms for Parallel & Distributed Systems
CS 391	Java & Web Design
CS 392	Computer Security
CS 393	Network Security
CS 394	Special Topics in Computer Science
CS 6843	Computer Networking
CS 2204	Digital Logic & State Machine Design
CS 3254	Introduction to Parallel & Distributed Systems
CS 3714	Secure Information Systems Engineering I
CS 3734	Scientific & Engineering Computing I
CS 4724	Secure Information Systems Engineering II
CS 4744	Scientific & Engineering Computing II
EE 136	Communication Networks
MA4423	Introduction to Numerical Analysis

Graduate Courses Open to Undergraduates

CS 6093	Advanced Database Systems
CS 6913	Web Search Engines
CS 6273	Performance Evaluation of Computer Systems
CS 6533	Interactive Computer Graphics
CS 6613	Artificial Intelligence I
CS 6643	Computer Visions & Scene Analysis
CS 6673	Neural Networking Computing
CS 6843	Computer Networking
CS 9013	Unix Systems
CS 9023	Applied Electronic Commerce
CS 9033	Programming Workshop (Algorithms & Software Libraries)
CS 9043	Cryptography with Financial Applications
CS 9053	Introduction to Java
CS 9073	Human & Computer Interaction
CS 9093	Computer Simulation
CS 9103	Object Oriented Design with Java
CS 9113	Machine Learning
CS 9133	Emerging Technology for IP Development
CS 9163	Application Security
CS 9663	Advanced Projectin Computer Science*
EL 5143	Multimedia Laboratory
EL 5473	Introduction to VLSI

* Offered every semester under different topics: information security management; penetration testing and system analysis; digital forensics; and cryptography.

GRADUATE PROGRAMS

GOALS AND OBJECTIVES MASTERS OF SCIENCE

The goals and objectives of the Master of Science in Computer Science program are to provide students with the following:

•Maximum curriculum flexibility, allowing students to adapt their program to their ambitions and goals as well as to their educational and professional backgrounds

•A solid grounding in the fundamentals of computer science

•Professional level courses in computer science

•Opportunity to specialize in selected technology areas of utmost interest

•Opportunities for a research-oriented program, in preparation for the Ph.D. program in computer science.

Master's Degree Requirements

To satisfy the requirements for the master's degree, the student must complete a total of 30 credits, as described below, with an overall average of B. In addition, a B average is required across the six core courses, as indicated below.

The master's curriculum has two components: 18 credits of core-elective courses and 12 credits of general elective courses.

Core-electives and Requirements

The core electives are organized into three core areas: Computer Systems, Programming/Software, and Theory. Students are required to take at least six core elective courses, with two courses coming from each of the core areas.

Systems Core Area:
CS 6133 Computer Architecture I
CS 6143 Computer Architecture II
CS 6233 Operating Systems I
CS 6243 Operating Systems II
CS 6253 Distributed Operating Systems
CS 6843 Computer Networking
CS 6813 Computer Security
CS 6823 Network Security

Theory Core Area CS 6003 Foundations of Computer Science
CS 6033 Design and Analysis of Algorithms I
CS 6043 Design and Analysis of Algorithms II
CS 6753 Theory of Computation
CS 6903 Modern Cryptography
CS 6703 Computational Geometry

Programming/Software Core Area:
CS 6063 Software Engineering I
CS 6073 Software Engineering II
CS 6083 Principles of Database Systems
CS 6373 Programming Languages
CS 6413 Compiler Design and Construction I
CS 6533 Interactive Computer Graphics
CS 6613 Artificial Intelligence I
CS 9163 Application Security

Additionally, for each of the courses Computer Architecture I (CS 6133), Operating Systems I (CS 6233), Design and Analysis of Algorithms I (CS 6033), Programming Languages (CS 6373), the following rule applies: The student is required to take the course unless the student has already taken an equivalent course (at either the graduate or undergraduate level) with a grade of B or higher; if the student has taken an equivalent course and received a B grade or higher, he/she will not be permitted to take it at Poly as part of this master's program without special permission.

General Electives Requirements

In addition to the core electives, students are required to take four general elective courses. There is considerable flexibility, with the only restriction being that no more than two of the courses being taken from outside

the Department of Computer and Information Science. In particular,

•Master's thesis (6 credits) and/or independent study courses may be part of a student's 4 elective courses
•Any of the courses in the three core areas may be chosen as electives.
•Graduate level courses from outside of the department (at most two) may be chosen as electives.
•Any CS graduate course not included in the core areas may be chosen as electives. These courses include:

CS 6273 Performance Evaluation of Computer Systems
CS 6643 Computer Vision & Scene Analysis
CS 6673 Neural Network Computing
CS 9013 Unix Systems (Perl)
CS 9053 Introduction to Java
CS 9073 Human & Computer Interaction
CS 6093 Advanced Database Systems
CS 9093 Computer Simulation
CS 9093 Biometric Identification
CS 9103 Object Oriented Design with Java
CS 6923 Machine Learning
CS 6913 Web Search Engines
CS 9133 Emerging Technology for IP Development

Preparatory Courses

The Department offers two preparatory bridge courses for students who do not have a working knowledge of a high level, general-purpose programming language:

CS 5303 Introduction to Programming
CS 5403 Data Structures and Algorithms

Master's Thesis

Exceptional students may elect to write a master's thesis, for which no more than 6 credits may be earned toward the degree. Such students should find an appropriate adviser who has agreed to monitor the thesis research. Such research need not be original, but should adequately demonstrate the student's proficiency in the subject material. An oral defense of the master's thesis with at least three professors in attendance is required.

PhD PROGRAM

Graduate students who have exhibited a high degree of scholastic proficiency and given evidence of ability for independent scholarly work may consider extending their goals toward the degree of doctor of philosophy. The preliminary requirements for admission to the program include the following:

1. A Bachelor's degree in science, engineering or management from an accredited school and a superior academic record, or
2. A Master's degree or one year of graduate work in an analytically-based area, and a superior academic record.

Applicants must include GRE general exam scores, at least two letters of recommendation, a statement of purpose, and all relevant academic records, in addition to the completed application form.

The Ph.D. program consists of 4 Parts:
A) Courses;
B) Qualifying exams;
C) Dissertation Proposal;
D) Dissertation.

Courses and Credits

A minimum of 75 credits of graduate work beyond the BS degree, including at least 21 credits of dissertation. A Master of Science degree in Computer Science may be transferred in as 30 credits without taking individual courses into consideration. Other graduate coursework may be transferred in on a course-by-course basis. This includes courses taken for degrees other than a Master of Science in Computer Science.

Students are required to take at least two courses of each of the following three areas. In the theory area, one of these two courses must be Theory of Computation (CS 6753) unless an equivalent course has already been taken. In selecting these courses, students are not to choose courses having substantial overlap with courses taken previously, at Polytechnic or elsewhere.

Systems Core Area:
CS 6143 Computer Architecture II
CS 6243 Operating Systems II
CS 6253 Distributed Operating Systems
CS 6843 Computer Networking
CS 6813 Computer Security
CS 6823 Network Security

Theory Core Area
CS 6043 Design and Analysis of Algorithms II
CS 6753 Theory of Computation
CS 6903 Modern Cryptography
CS 6703 Computational Geometry

Programming/Software Core Area:

CS 6063 Software Engineering I
CS 6073 Software Engineering II
CS 6083 Principles of Database Systems
CS 6413 Compiler Design and Construction I
CS 6533 Interactive Computer Graphics
CS 6613 Artificial Intelligence I
CS 9163 Application Security

The CIS faculty may modify these area menus from time to time. Certain selected topics courses can be used to fulfill these requirements, with prior written permission from the CIS Department.

Students must receive at least a grade of B in each of the six courses; further, the grade point average averaged over these six courses must be at least a 3.5. Full-time students must complete these course requirements by the end of their second year.

Additionally, for each of the courses Computer Architecture I (CS 6133), Operating Systems I (CS 6233), Design and Analysis of Algorithms I (CS 6033), Programming Languages (CS 6373), the following rule applies: The student is required to take the course unless the student has already taken an equivalent course (at either the graduate or undergraduate level)

with a grade of B or higher; if the student has taken an equivalent course and received a B grade or higher, he/she will not be permitted to take it at Poly as part of this Ph.D. program without special permission.

Qualifying Exam

The qualifying exam will be offered once a year. Students must take the exam for the first time before the end of their second year in the program. In general, a student will pass or fail the exam in its entirety; however, exceptions may be made in rare cases. Failure of the exam twice will result in immediate dismissal from the program. For a description of the exam, consult the department handbook.

Students cannot register for dissertation credit until they have passed the qualifying exam.

Dissertation Proposal

After passing the qualifying exam, students should begin conducting research.

The student, in consultation with their advisor, forms a dissertation committee of at least 4 faculty members, including one member from another department or from outside the University. At least six months before the dissertation defense, the student is to prepare a dissertation proposal and orally present the proposal to the dissertation committee.

Dissertation

The last, and most substantial, aspect of the Ph.D. program is the dissertation. The dissertation must embody a significant original research contribution and must be written in accepted scholarly style. The research should be conducted in close consultation with the student's advisor. It is strongly recommended that at least one paper on the research be submitted to a refereed archival journal or refereed conference. When the adviser feels that sufficiently significant research results have been obtained and that the dissertation has been written in an acceptable way, a public dissertation defense, consisting of an oral presentation by the candidate and questions by the dissertation committee, will be scheduled.

Additional requirements for the Ph.D. dissertation are available from the office of the Dean of Graduate Studies.

CERTIFICATES IN INFORMATION ASSURANCE

As a National Security Agency-designated Center of Academic Excellence in Information Assurance, Polytechnic is offering NSA–approved certificates in information assurance. The certificates are awarded to students who are pursuing a bachelor's or master's degree in computer science, computer engineering, telecommunication or electrical engineering at the University and have completed the following course requirements:

NSTISSI 4011: Information Security Professional
Course Requirements for the Information Security Professional Certificate (27 Credits)

CS 5403 or
CS 2134 Data Structures and Algorithms*
CS 6133
CS 2214 Computer Architecture*
CS 6233 or
CS 3224 Operating Systems*
CS 6373 or
CS 3314 Programming Languages*
CS 6843 Computer Networks**
CS 6813 Information, Privacy and Security
or
CS 392 Computer Security
CS 6823 Network Management and Security
or
CS 393 Network Security
EL 5363 or
EE 136 Principles of Communication Networks*
CS 6804 Information Security Management

NSTISSI 4013: Information Systems Administration
Course Requirements for the Information Systems Administration Certificate

4011: Information Systems Administration Certificate plus
CS 6573 Penetration Testing and System Analysis
CS 6243 Operating Systems II***

Note: ∗ These courses can be waived if the student has attended a comparable course and has demonstrated proficiency in the required topics/subjects.
∗∗ Can be replaced with EL 5373
∗ ∗ ∗ Can be replaced with CS 3253

For more information, contact Professor Nasir Memon at memon@poly.edu.

GRADUATE CERTIFICATE IN CYBER SECURITY

The demand for skilled information security professionals is growing significantly. This graduate certificate allows technical professionals to obtain key bodies of knowledge and specializations in cyber security. Students acquire an understanding of various technologies in emerging areas of security, including computer and network security, digital forensics, cryptography and biometrics. Students are able to immediately apply their knowledge to manage the risk of cyber attacks. Courses are developed and taught by Polytechnic faculty in the Information Systems and Internet Security (ISIS) Laboratory. Those who choose to work toward a master's degree are able, upon admission, to apply all courses taken toward a certificate toward fulfillment of a degree program. Admission to the certificate program requires a bachelor's degree in an appropriate preparatory discipline from an institution acceptable to Polytechnic.

Course Requirements for the Cyber Security Certificate (15 credits)

Core Courses 9 credits

CS 6803* Information Systems Engineering & Management
CS 6813* Information, Security & Privacy
CS 6823* Network Management & Security

Electives 6 units

CS 9093* Biometrics
CS 6903 Modern Cryptography
CS 9163* Application Security
CS 6963* Digital Forensics
* = available online.

For more information, contact Professor Nasir Memon at memon@poly.edu

GRADUATE CERTIFICATE IN SOFTWARE ENGINEERING:

The advanced certificate in software engineering consists of a series of five graduate level courses designed to give students the knowledge and skills they need to compete successfully in the software development arena. Students who want to continue in their studies can apply these courses to the ms program in computer science.

In response to the importance of high quality software development and integration industry, Polytechnic offers a certificate program in software engineering. This course module gives the students the knowledge and skills needed to compete successfully in this arena. Topics covered include object oriented software design, software validation and project management.

The software engineering certificate is a series of five graduate-level courses. Three required core courses are designed to equip the computer science professional for an advanced software development career. In addition, students choose two other courses from a variety of electives offered on a rotating basis. The elective courses cover areas of current interest to the software engineering community and allow students to customize their education.

Core Courses 9 credits

CS 6063 Software Engineering I
CS 6073 Software Engineering II
CS 6083 Principles of Database Systems

Electives 6 credits

CS 9963 Advanced Project in Computer Science*
CS 9103 Object Oriented Design with Java
CS 9163 Application Security
CS 6813 Information Privacy and Security

*Highly recommended

Entrance requirements for the certificate program are the same as for the MS program. For students having superior academic credentials but lacking sufficient background in computer science, there are two prerequisite courses (CS 5303 Introduction to Programming Methodology, and CS 5403 Data Structures and Algorithms).

UNDERGRADUATE COURSES

Students are advised to consult the Schedule of Classes for changes in prerequisites in effect after publication of this catalog. Students may not register for any junior- or senior-level courses until all freshmen requirements are completed. The annotation 3:0:0:3 means that the course meets for 3 lecture hours, 0 laboratory hours and 0 recitation hours each week and that a total of 3 credits (or units for graduate courses) are awarded upon successful completion of the course.

CS 205 Assembly Language and Systems Programming

3:0:0:3

Internal representation of numeric and character data. Machine organization and machine language programming. Assembly language, assemblers. Assembly language programming: branching, arrays, lists, arithmetic and bit manipulation, macros, stacks, subroutines, parameter passing, recursion. Linking and loading, position-independent and reentrant code. Traps and interrupts. *Prerequisite: CS 2134 (C- or better).*

CS 239 UNIX System Programming

3:0:0:3

Programming and system administration of UNIX systems. Covers shell programming, special purpose languages, UNIX utilities, UNIX programming tools, systems programming and system administration. *Prerequisites: CS 3224 and junior status.*

CS 308 Introduction to Databases

3:0:0:3

This course introduces students to database systems and motivates the database approach as a mechanism for modeling the real world. The course will cover data models (relational, object-oriented), physical database design, query languages, query processing and optimization, as well as transaction management techniques. Implementation issues, object-oriented and distributed databases will also be introduced. *Prerequisites: CS 2134 and CS 3224.*

CS 342 Algorithms for Parallel and Distributed Systems

3:0:0:3

Covers the design, implementation and evaluation of algorithms for parallel and distributed systems. Scheduling and loadbalancing, parallel and distributed information retrieval and database operations, parallel scientific algorithms. Concurrency control. Security in distributed systems. *Prerequisite: CS 3254.*

CS 391 Java and Web Design

3:0:0:3

Programmers familiar with C or C++ will learn how to develop Java applications and applets. This course will teach students the syntax of the Java language, object-oriented programming in Java, creating graphical user interfaces (GIU) using the Java 2 Platform technology event model, Java exceptions, file input/output (I/O) using Java Foundation Class threads and networking. *Prerequisite: CS 2134.*

CS 392 Computer Security

3:0:0:3

Covers the following topics: Cryptographic systems. Capability and access-control mechnanisms, authentication models, protection models. Database and operating system security issues, mobile code, security kernels. Malicious code, trojan horses and computer viruses. Security policy formation and enforcement, legal aspects and ethical aspects. *Prerequisites: CS 2214 and MA 2312. Co-requisite: CS 3224*

CS 393 Network Security

3:0:0:3

Review of topics in networking. Basic notations of confidentiality, integrity, availability; cryptographic systems, coding and decoding messages. cryptographic protocols for privacy, integrity, key exchange and access control. TCP/IP security; Firewalls, IPSec; secure e-commerce. Intrusion detection, prevention, response. Advanced topics. *Prerequisites: CS 3224 and CS 684, or EE 136, EL 536 or EL 537.*

CS 394 Special Topics in Computer Science

A variable credit special topics course designed for juniors and seniors. *Prerequisite: department's permission.*

CS 1114 Introduction to Programming and Problem Solving

3:3:0:4

An introduction to computer programming and problem solving. General topics covered include the fundamentals of programming, good software development practices and solving problems using computer programming. Specific topics include compiling, running and debugging a program, program testing, documentation, variables and data types, assignments, arithmetic expressions, input and output, top-down design and procedures, the random number generator, conditionals and loops functions, arrays, and an introduction to classes and object oriented programming. Grade of C- or better required of undergraduate computer science and computer engineering majors.

CS 1124 Object-Oriented Programming

3:3:0:4

An intermediate-level programming course teaching object-oriented programming in C++. Pointers, dynamic memory allocation, and recursion. Classes and objects including constructors, destructors, methods (member functions) and data members. Access and the interface to relationships of classes including composition, association, and inheritance. Polymorphism through function overloading operators. Inheritance and templates. The standard template library will be used to introduce elementary data structures and their use. Grade of C- or better required of computer science and computer engineering majors. *Prerequisite: CS 1114 (C- or better).*

CS 2102 Computer Science for Civil Engineers

2:0:0:2

The objective fo the course is to introduce Civil Engineering students ot computer programming. The course will emphasize engineering problem solving through the use of the Java language. Students will be exposed to the concept of compiling, debugging, and writing Java progrmas to solve problems. Problems related to Civil Enindeering student will be emphasized. *Credit will not be granted for both CS 2102 and CS 1114, Prerequisite: major in a Civil Engineering discipline*

CS 2113 Programming with VBA/Excel

2.5:0:0:3

This is an introductory course in computer programming and problem solving for undergraduate students in the Biology/Molecular Science program who have no prior experience in programming in any language. The course covers the fundamentals of computer programming and its underlying principles using the programming language Visual Basic for Applications (VBA). Subroutine and function procedures are created to be run within the Excel environment. This course is only for Biology/Molecular Science Students. *Prerequistes: None*

CS 2134 Data Structures and Algorithms

4:0:0:4

Abstract data types and the implementation and use of standard data structures. Fundamental algorithms and the basics of algorithm analysis. Grade of C- or better required of undergraduate computer science and computer engineering majors. *Prerequisites: CS 1124 (C- or better) and MA 1024. Co-requisite: MA 2312/2322.*

CS 2204 Digital Logic and State Machine Design

3:3:0:4

Combinational and sequential digital circuits. An introduction to digital systems. Number systems and binary arithmetic. Switching algebra and logic design. Error detection and correction. Combinational integrated circuits, including adders. Timing hazards. Sequential circuits, flipflops, state diagrams and synchronous machine synthesis. Programmable Logic Devices, PLA, PAL and FPGA. Finitestate machine design. Memory elements. Grade of C- or better required by undergraduate computer engineering majors. *Prerequisite: CS 1114 (C- or better).*

CS 2214 Computer Architecture and Organization

3:3:0:4

A top-down approach to computer design. Computer architecture: introduction to assembly language programming and machine language set design. Computer organization: logical modules; CPU, memory and I/O units. Instruction cycles, the datapath and control unit. Hardwiring and microprogramming. The memory subsystem and timing. I/O interface, interrupts, programmed I/O and DMA. Introduction to pipelining and memory hierarchies. Fundamentals of computer networks. *Prerequisites: CS 2204 and CS 2134 (C- or better for undergraduate computer engineering majors); CS 2134 (C- or better) and MA 2312/2322 for CS students.*

CS 3224 Operating Systems

4:0:0:4

This course will study the fundamental concepts and principles of operating systems. Batch, spooling, and multiprogramming systems are introduced. The parts of an operating system are described in terms of their functions, structure and implementation. Basic policies for allocating resources are also discussed. *Prerequisite: CS 2214.*

CS 3254 Introduction to Parallel and Distributed Systems

3:3:0:4

The goal of this course is to give you a solid grounding in the basic issues and techniques of parallel and distributed computing. The material we cover will cover the spectrum from theoretical models of parallel and distributed systems to actual programming assignments. *Prerequisite: CS 2134.*

CS 3314 Design and Implementation of Programming Languages

4:0:0:4

This course covers issues underlying the design of high-level programming languages, along with elements of the compiler technology used to translate those languages into executable code. Topics covered include formal description of language syntax, parsing, memory management, attributes of variables and their binding times, control and data abstraction mechanisms, and object-oriented language features. The focus is on imperative and object-oriented languages, with brief introduction to functional and logicprogramming paradigms. Substantial programming projects are required. *Prerequisites: CS 2134 (C- or better) and MA 2312.*

CS 3414 Design and Analysis of Algorithms

4:0:0:4

This course covers the fundamental principles of the design and analysis of algorithms. Topics include asymptotic notation, recurrences, randomized algorithms, sorting and selection, balanced binary search trees, augmented data structures, advanced data structures, algorithms on strings, graph algorithms, geometric algorithms, greedy algorithms, dynamic programming, and NP completeness. *Prerequisites: CS 2134 (C- or better) and MA 2312/2322.*

CS 3714 Secure Information Systems Engineering I

4:0:04

An approach to secure information systems engineering is developed consistent with today's vulnerabilities, threats and risks. Grounding is established in the basic security technologies and strategies in use today. A concept of security engineering is constructed for whole elements of the critical infrastructure (e.g., utilities, government services, financial services, etc.) including legacy environments, the Internet, wireless and the coming evolution of "ubiquitous computing." *Prerequisite: junior standing.*

CS 3734 Scientific and Engineering Computing I

4:0:04

This course takes advantage of the programming skills that students learn in introductory level computer science courses to exploit the broad power of modern computing related to their science and engineering disciplines. Computational techniques are taught in parallel with programming and problemsolving methodologies. Students learn how to recognize a good or bad formulation of a problem, select the proper algorithm to solve a given computational problem and interpret the results; thus, learning to become intelligent users, rather than creators, of computational software. *Prerequisites: CS 1114, MA 1124, MA 2012/2132.*

CS 4513 Software Engineering I

2 1/2:11/2:0:3

The first in a two-course design project sequence (DP I and DP II) with a focus in software engineering. This course introduces the software engineering techniques to specify, design, test and document medium and large software systems. Design techniques include information engineering, object-oriented, and complexity measures. Testing methods such as path testing, exhaustive test models, and construction of test data. An introduction to software tools and project management techniques is presented. Student projects involve team software development and tracking. *Prerequisites: CS 2134 (C- or better), CS 3224 and senior status.*

CS 4523 Design Project II

1:6:0:3

The second course in a two-course design project sequence (DP I and DP II). This is a project course in which a student or several students work with a faculty member and/or graduate students on a current topic in computer science. Each term, a project course with a particular theme is offered by the Department of Computer and Information Science. A faculty member will assign individual or group projects to students in the class. The project course will be highly structured and will be under close supervision of the faculty. It is expected that students will make use of the design and project management skills they have learned in CS 4513 Software Engineering. Alternatively, students can work with a faculty member to develop an individual project of mutual interest. A written report and oral presentation are required. *Prerequisite: CS 4513.*

CS 4724 Secure Information Systems Engineering II

4:0:04

The second semester consists of projects, labs and discussions in the area of Applied Secure Information Systems Engineering. Specifically, students build a comprehensive platform for secure computing based on best of breed Open Source components starting with OpenBSD or the like. This platform is then contrasted to Java Security and to the secure computing efforts of Microsoft and its associates. A "Student Hackathon" is conducted to test findings and assumptions. Finally, recommendations are made to support the future security procurement needs for whole elements of the critical Infrastructure.

CS 4744 Scientific and Engineering Computing II

4:0:04

Making use of the knowledge acquired in part I of the course, the second semester focuses on well recognized major computational developments that have the greatest influence on the development and practice of science and engineering in the last century. Course draws upon a variety of computational problems from the breadth of science and engineering to interest students and establish the relevance of the computational problem-solving approach. Students will be involved in projects. *Prerequisite: CS 3734.*

GRADUATE PREPARATORY COURSES

The graduate courses listed in this section were formulated to accommodate the needs of students who wish to pursue graduate studies in computer science, but who lack sufficient undergraduate preparation. No credit will be allowed for any of these courses toward graduate degrees in computer science, information systems engineering or other graduate degree programs administered by the Department of Computer and Information Science. Submission of substantial computer programming assignments is required in all these courses.

CS 5303* Introduction to Programming and Problem Solving

3:0:0:3

Provides an introduction to discrete mathematics, computers and programming; running C/C++ programs under Unix; algorithmic language; pseudocode; problem solving and program structure. Topics covered include constants, variable, data types, assignments, arithmetic expressions, input and output; object - oriented and top down design and procedures, selection and loops; functions; enumerated; arrays, structs and searching and sorting. *Prerequisite: graduate status : Online version available*

CS 5403* Data Structures and Algorithms

3:0:0:3

This course provides an introduction to data structures. Topics include; program specifications and design; abstract data types; stacks, queues; dynamic storage allocation; sequential and linked implementation of stacks and queues; searching methods, sequential and binary; binary trees and general trees; hashing; computational complexity; sorting algorithms selection sort, heap sort, merge-sort, and quick-sort; comparison of sorting techniques and analysis. *Prerequisite: CS 5303 : Online version available*

CS 580* Computer Architecture and Organization

3:0:0:3

This course provides computer science students with an understanding of computer hardware subsystems, and basic digital design tools and strategies. Combinational and sequential circuits are developed for the basic building blocks of computers. Binary and hexadecimal arithmetic is presented in both human and machine algorithms. A simple computer

is built up from the building blocks developed. Hardwired and microprogrammed control systems are investigated. Assembly language and instruction sets are presented. Memory organization alternatives are explored. Prerequisite: graduate status. Co-requisite: CS 530. *: Online version available*

CS 590* Introduction to Operating Systems

3:0:0:3

This course describes operating systems, the programs that interface with computer hardware. These programs can be implemented in firmware or software. The concept of process as program in execution is introduced, and operations (suspend and resume) on processes discussed. Hardware and software mechanisms for providing asynchronous processes with mutually exclusive access to resources and for avoiding deadlocks are given. Scheduling strategies for CPUand IO-bound process are described. Storage management is presented as the second major component of an operating system. The organization of physical storage is discussed. The implementation of virtual memory and file systems is described. The importance of security and protection in multi-user systems is discussed. A case study may be presented. Prerequisites: CS 530 and CS 580. *: Online version available*

GRADUATE COURSES

Graduate courses in computer science are offered on each campus on a regular basis, annually, or in two- or three-year cycles. (3:0:0:3) a total of 3 credits are awarded upon successful completion of the course.

CS 6003 Foundations of Computer Science

3:0:0:3

Logic, sets, functions, relations, asymptotic notation, proof techniques, induction, combinatorics, discrete probability, recurrences, graphs, trees, mathematical models of computation, undecidability *Co-requisite: CS 5303.*

CS 6033 Design and Analysis of Algorithms I

3:0:0:3

Review of basic data structures and mathematical tools. Data structures: priority queues, binary search trees, balanced search trees. B-trees. Algorithm design and analysis techniques illustrated in searching and sorting: heapsort, quicksort, sorting in linear time, medians and order statistics. Design and analysis techniques: dynamic programming, greedy algorithms. Graph algorithms: elementary graph algorithms (breadth-first search, depth-first search, topological sort, connected components, strongly connected components), minimum spanning tree, shortest path. String algorithms. Geometric algorithms. Linear programming. Brief introduction to NP-completeness. *Prerequisites: CS 5403 and CS 6003.*

CS 6043 Design and Analysis of Algorithms II

3:0:0:3

Advanced design and analysis techniques: amortized analysis of algorithms. Advanced data-structures: binomial heaps, Fibonacci heaps, data structures for disjoint sets, analysis of union by rank with path compression. Graph algorithms: elementary graph algorithms, maximum flow, matching algorithms. Randomized algorithms. Theory of NPcompleteness and approach to finding (approximate) solutions to NP-complete problems. Selected additional topics that may vary. *Prerequisite: CS 6033.*

CS 6063 Software Engineering I

3:0:0:3

The emphasize is on understanding the full software engineering approach with alternative approaches. Technical emphasis is on requirements, design, development, and modeling. Management issues include software cost estimating and project management. Understanding the processes applicable to the software development/integration cycle and maintenance along with technology changes on quality and development activities is highlighted. *Prerequisite: CS 5403.*

CS 6073 Software Engineering II

3:0:0:3

: Covers modern advanced software engineering approaches of both a theory and practice orientation. Important Design and Management issues are analyzed and evaluated. Technical and management tradeoffs in distributed software systems are emphasized. An Extensive number of real world case studies are assessed. A class project is required. *Prerequisite: CS 6063.*

CS 6083 Principles of Database Systems

3:0:0:3

This course aims to give a broad introduction to database systems, including the relational data model, query languages, database design, index and file structures, query processing and optimization, concurrency and recovery, transaction management, and database design. Students will also acquire hands-on experience in working with database systems and in building web-accessible database applications *Prerequisites: CS 3083 or instructor's permission, and CS 6003.*

CS 6093 Advanced Database Systems

3:0:0:3

This course aims to give a broad introduction to database systems, including the relational data model, query languages, database design, index and file structures, query processing and optimization, concurrency and recovery, transaction management, and database design. Students will also acquire hands-on experience in working with database systems and in building web-accessible database applications

CS 6133* Computer Architecture I

3:0:0:3

This course provides students with an understanding of computer hardware subsystems, digital design strategies and fundamental computer performance and capacity improvement techniques. Combinational and sequential circuits are developed for the essential building blocks of computers. Binary number systems are presented in both human and computer algorithms. A uniprocessor computer is built up from the blocks developed. An assembly language and an instruction set are presented. Processor implementation with a data path and hardwired and microprogrammed control is introduced. Performance evaluation of computers is studied. Basic pipelining is introduced to improve the performance of the system. Memory hierarchy alternatives are introduced to improve the capacity of the computing system. *Prerequisite: CS 5803 : Online version available*

CS 6143 Computer Architecture II

3:0:0:3

High-speed computer arithmetic. Uniprocessor computer architectures that exploit parallelism: Advanced pipelining, superscalar, VLIW, vector processors. Parallel processing: Interconnection structures, MIMD and SIMD systems. Other selected parallel computing topics, such as, parallel algorithms, PRAM machines and multi-core processing. *Prerequisite: CS 6133.*

CS 6183 Fault-Tolerant Computers

3:0:0:3

Introduces a variety of hardware and software techniques for designing and modeling fault-tolerant computers. Topics include coding techniques (Hamming, SECSED, SECDED, etc.); majority voting schemes (TMR); software redundancy (N-version programming); software recovery schemes; network reliability design and estimation. Introduces probabilistic methods for reliability modeling. Examples from space fault-tolerant systems, networks, commercial nonstop systems (TANDEM and STRATUS). RAID memory systems. Fault-tolerant modeling tools such as HARP, SHURE and SHARPE. *Prerequisites CS 6133*

CS 6233 Introduction to Operating Systems

3:0:0:3

Introduction to basic issues in operating systems. Threads, Processes, Concurrency, Memory Management, I/O Control and case studies. *Prerequisite: graduate standing*

CS 6243 Operating Systems II

3:0:0:3

Survey of recent important commercial and research trends in Operating Systems. Topics may include virtualization, network server design and characterization, scheduling and resource optimization, file systems, memory management, advanced debugging techniques, data center design, and energy utilization. *Prerequisite: CS 6233.*

CS 6253 Distributed Operating Systems

3:0:0:3

Introduction to distributed networked computer systems. Distributed control and consensus. Notions of time in distributed systems. Client-Server communications protocols. Middleware. Distributed File Systems and Services. Fault tolerance, replication and transparency. Peer to peer systems. Case studies of modern commercial systems and research efforts.

CS 6273 Performance Evaluation of Computer Systems

3:0:0:3

This course focuses on modeling and performance analysis of computer systems. In particular this course will concentrate on Testing and Evaluation of three tiered Distributed Client/Server and WEB Based Systems and more generally distributed networking systems. The course will present and evaluate various systems architectures from a macro and micro viewpoint. *Prerequisites: EL 5313 or MA 2212/2222 and instructor's permission.*

CS 6373 Programming Languages

3:0:0:3

The structures, notations, and semantics of programming languages. Issues of scope, type structure, and parameter passing. Control structures, including support for exception handling and concurrency. Abstract data types and object-oriented languages. Programming in the large. Implementation issues. Functional, logic programming languages. Examples from a variety of languages. *Prerequisites: CS 5403.*

CS 6413 Compiler Design and Construction

3:0:0:3

Compiler organization. Lexical analysis, syntax analysis, abstract syntax trees, symbol table organization, code generation. Introduction to code optimization techniques. *Prerequisites: CS 5403, CS 5803 and CS 6003.*

CS 6533 Interactive Computer Graphics

3:0:0:3

This course introduces the fundamentals of Computer Graphics with hands-on graphics programming experiences. Topics include: graphics software and hardware, 2D line-segment scan conversion, 2D and 3D transformations, viewing, clipping, polygon scan-conversion, hidden-surface removal, illumination and shading, compositing, texture mapping, ray tracing and radiosity, scientific visualization, and so on. *Prerequisites: CS 5403 (Data Structures) or equivalents, and knowledge of C or C++ programming*

CS 6573* Penetration Testing and Vulnerability Analysis

3:0:0:3

This is an advanced course in computer and network security that focuses on penetration testing and vulnerability analysis. The course introduces various methodologies, techniques and tools to analyze and identify vulnerabilities in stand–alone and networked applications. *Prerequisites: CS 6823 : Online version available*

CS 6613 Artificial Intelligence I

3:0:0:3

Artificial Intelligence (AI) is an important topic in computer science that has many diversified applications. It addresses one of the ultimate puzzles human are trying to solve How is it possible for a slow, tiny brain, whether biological or electronic, to perceive, understand, predict, and manipulate a world far larger and more complicated than itself? And, how do we go about creating a machine (or computer) with those properties? To this end, researchers in the AI field have been trying to understand how seeing, learning, remembering, and reasoning could, or should be done. This course introduces students to the many concepts and techniques in artificial intelligence. *Prerequisite: CS 5403.*

CS 6643 Computer Vision and Scene Analysis

3:0:0:3

An important goal of artificial intelligence is to equip computers with the capability of interpreting visual inputs. Computer vision and scene analysis is an area in AI that deals with the construction of explicit, meaningful descriptions of physical objects from images. It includes as parts many techniques from image processing, pattern recognition, geometric modeling, and cognitive processing. This course introduces students to the many techniques and applications of computer vision and scene analysis. *Prerequisites: CS 5403 and MA 2012 or equivalents or instructor's permission.*

CS 6673 Neural Network Computing

3:0:0:3

An introduction to neural network models and their applications. Discussion of organization and learning in neural network models including perceptrons, adalines, backpropagation networks, recurrent networks, adaptive resonance theory and the neocognitron. Implementations in general and special purpose hardware, both analog and digital. Application in various areas with comparisons to non-neural approaches. Decision systems, nonlinear control, speech processing and vision. *Prerequisite: CS 5403. Some familiarity with matrix notation and partial derivatives is recommended.*

CS 6703 Computational Geometry

3:0:0:3

This course will present and introduction to data structures and algorithms for geometric data. Topics will include: line-segment intersection, polygon triangulation, linear programming, orthogonal range searching, point location, Voronoi diagrams, Delaunay triangulations, arrangements and duality, geometric data structures, convex hulls, binary space partitions, robot motion planning, quadtrees, visibility graphs, simplex range searching.

CS 6753 Theory of Computation

3:0:0:3

Introduction to the theory of computation. Formal languages and automata theory. Deterministic and non–deterministic finite automata, regular expressions, regular languages, context–free languages. Pumping theorems for regular and context–free languages. Turing machines, recognizable and decidable languages. Limits of computability: the Halting Problem, undecidable and unrecognizable languages, reductions to prove undecidability. Time complexity, P and NP, Cook–Levin theorem, NP–completeness. *Prerequisites: CS 6003 or instructor's permission.*

CS 6803* Information Systems Security Engineering & Management

3:0:0:3

The primary goal of this course is to present a system and management view of information security: what it is, what drives the requirements for information security, how to integrate it into the systems design process, and life cycle security management of information systems. A second goal is to cover basic federal government information security policies and methodologies. Topics covered include information security risk management, security policies, security in the systems engineering process, laws related to information security, and management of operational systems. *Prerequisite: CS 392 or equivalent : Online version available*

CS 6813* Information, Security and Privacy

3:0:0:3

This is an introductory course in Information Systems Security that deals with following topics: Cryptography, capability and access-control mechanisms, authentication models, security models, operating systems security, malicious code, security policy formation and enforcement, vulnerability analysis, evaluating secure systems. *Prerequisite: graduate status : Online version available*

CS 6823* Network Security

3:0:0:3

This course first covers attacks and threats in computer networks, including network mapping, port scanning, sniffing, DoS, DDoS, reflection attacks, attacks on DNS, and leveraging P2P deployments for attacks. The course then covers the topics in cryptography that are most relevant to secure networking protocols. These topics include block ciphers, stream ciphers, public-key cryptography, RSA, Diffie-Hellman, certification authorities, digital signatures, and message integrity. After surveying the basic cryptographic techniques, the course examines a number of secure networking protocols, including PGP, SSL, IPsec, and wireless security protocols. The course also examines operational security, including firewalls and intrusion detection systems. The course involves reading some recent research papers on network security. The course also has an important lab component, including labs on packet sniffing, network mapping, firewalls, SSL, and IPsec. *Prerequisite: EL 5363 : Online version available*

CS 6843 Computer Networking

3:0:0:3

This course takes a top-down approach to computer networking. After providing an overview of computer networks and the Internet, the course covers the application layer, transport layer, network layer and link layers. Topics at the application layer include client-server architectures, P2P architectures, DNS, and HTTP and Web applications. Topics at the transport layer include multiplexing, connectionless transport and UDP, principles or reliable data transfer, connection-oriented transport and TCP, and TCP congestion control. Topics at the network layer include forwarding, router architecture, the IP protocol, and routing protocols including OSPF and BGP. Topics at the link layer include multiple access protocols, ALOHA, CSMA/CD, Ethernet, CSMA/CA, wireless 802.11 networks, and link-layer switches. The course includes simple quantitative delay and throughput modeling, socket programming and network application development, and Ethereal labs. *Prerequisite: CS 2134.*

CS 6873 Project in Telecommunication Networks

2 1/2:0:0:3

A design course where students design, develop and test communication software. It is expected that the student will work in small groups under the direction of a professor. Students will have access to network resources for their work. *Prerequisites: CS 6843 and instructor's permission.*

CS 6903 Modern Cryptography

3:0:0:3

This course deals with the study of modern cryptography from a theoretical perspective, the emphasis of the course being on "provable security". In particular, we study the cryptographic primitives that are the building-blocks of various cryptographic applications. The course involves the study of notions of security for a given cryptographic primitive, its various constructions, and respective security analysis based on the security notion. The cryptographic primitives that we cover include pseudorandom functions, symmetric encryption (block ciphers), hash functions and random oracles, message authentication code, asymmetric encryption and digital signatures. Time permitting; we also study how to build secure cryptographic protocols for authenticated key exchange, using the primitives that we study. We also study various number-theoretic assumptions cryptography is based upon.

CS 6913 Web Search Engines

3:0:0:3

This course covers the basic technology underlying web search engines and related tools. The main focus will be on large-scale web search engines (such as Google, Yahoo, and MSN Search) and the underlying architectures and techniques. Students will learn how search engines work, and get hands-on experience in how to build search engines from the ground up. Topics are taught based on a reading list of recent research papers. Students have to work on a course project and may have to present in class.

CS 6923 Machine Learning

3:0:0:3

Introduction to the field of machine learning Standard machine learning techniques such as decision trees, nearest neighbor, Bayesian methods, support vector machines, and logistic regression. Basic concepts in computational learning theory including the PAC model and VC dimension. Methods for evaluating and comparing machine learning techniques.

CS 6963* Digital Forensics

3:0:0:3

This course will introduce information technology professionals with the application of forensic science principles and practices to the collection, preservation, examination, analysis and presentation of digital evidence. The course will include selected topics from the legal, forensic, and information technology domains and utilize lecture, laboratory and written projects to illustrate these topics. *Online version available*

CS 9013, CS 9023, CS 9033, ···, CS 9253 Selected Topics in Computer Science

each 3 credits

Topics of current interest in computer science. Recent offerings include software specification and validation, parallel algorithms and architectures, client server systems and advanced object oriented design (Java). Advanced topics in databases, performance analysis, computer simulation, Java programming, Unix programming, human and computer interaction, cryptography with financial applications and biometric identification. *Prerequisites: specified when offered.*

CS 9093* Biometrics

3:0:0:3

The course concentrates on the unique advantages that biometrics brings to computer security, but also addresses challenging issues such as security strength, recognition rates, and privacy, as well as alternatives of passwords and smart cards. Students will gain knowledge in the building blocks of this field: image and signal processing, pattern recognition, security and privacy, and secure system design. By the end of the course students will be able to evaluate and design security systems that include biometrics. *Online version available*

CS 9163* Application Security

3:0:0:3

This course addresses the designing and implementation of secure applications. Concentration is on writing software programs that make it difficult for intruders to exploit security holes. The course will have emphasis on writing secure distributed programs in Java. The security ramifications of class, field and method visibility and the exploration of important sec more robust will be emphasized. *Online version available*

CS 9413/CS 9423 Readings in Computer Science I/II

each 3 credits

Intended primarily for advanced graduate students who wish to study in a specialized area under the supervision of a faculty member. Permission of graduate director is required. Regular meetings with the adviser. Examination or term report required. *Prerequisite: graduate status.*

PROJECT AND THESIS

Students may register and get credit for these courses more than once.

CS 9963 Advanced Project in Computer Science

3:0:0:3

This course permits the student to perform research in computer science with a narrower scope than a master's thesis. The acceptance of a student by a faculty adviser is required before registration. An oral examination on the project report is required. *Prerequisite: graduate status.*

CS 9973 Thesis for Degree of Master of Science

3 credits each

Exceptional students may elect to write a master's thesis for which no more than 6 units may be earned toward the degree. Such research should adequately demonstrate the student's proficiency in the subject material. Oral thesis defense with at least three professors in attendance plus a formal, bound thesis volume are required. Thesis registration must be continuous. *Prerequisites: graduate status and satisfactory grades in prescribed courses.*

CS 9983 Variable Credit Project/Course

variable 3 each

For students needing .5, 1, 1.5 and 2 credit hours to meet graduation requirements, a project or special course is available with Faculty approval

CS 9993 Dissertation for Degree of Doctor of Philosophy

each 3 credits

Original investigation of computer science problem. Must demonstrate creativity and include features of originality and utility worthy of publication in a recognized journal. Candidate must successfully defend dissertation orally. Registration of 24 units required (continuous dissertation registration required). *Prerequisites: passing of qualifying examination and approval of the Department of Computer and Information Science.*

Typical Course of Study for the Bachelor of Computer Science

FRESHMAN YEAR

Fall Semester — *Hour/Week*

CourseNo.	*CourseTitle*	*Class*	*Lab*	*Rec.*	*Cr*
CS 1114	*Intro. Prog. & Problem Solving*[1]	3	3	0	4
CM 1004	*General Chemistry*	3	2	1	4
EN 1014	*Writing & Humanities I*[2]	4	0	0	4
MA 1024	*Calculus I*	4	0	0	4
SL 1010	*Freshman Seminar*	1	1	0	0
					16

Spring Semester — *Hours/Week*

CourseNo.	*CourseTitle*	*Class*	*Lab*	*Rec.*	*Cr*
CS 1124	*Object – Oriented Programming*	3	3	0	4
EG 1004	*Intro. Engineering & Design*	1	3	2	4
MA 1124	*Calculus II*	4	0	0	4
EN 1204	*Writing & Humanities II*	4	0	0	4
					16

SOPHOMORE YEAR

Fall Semester — *Hour/Week*

CourseNo.	*CourseTitle*	*Class*	*Lab*	*Rec.*	*Cr*
CS 2134	*Data Structures & Algorithms*	4	0	0	4
MA 2312	*Discrete Math. I (1/2 semester)*	4	0	0	2
MA 2322	*Discrete Math. II (1/2 semester)*	4	0	0	2
PH 1004	*Introductory Physics I*	4	1 1/2	1	4
HI 2104	*Modern World History*	4	0	0	4
					16

Spring Semester — *Hours/Week*

CourseNo.	*CourseTitle*	*Class*	*Lab*	*Rec.*	*Cr*
CS 2214	*Computer Arch. & Organization*	3	3	0	4
MA 2212	*Data Analysis I (1/2 semester)*	4	0	0	2
MA 2222	*Data Analysis II (1/2 semester)*	4	0	0	2
PH 2004	*Introductory Physics II*	4	1 1/2	1	4
	HU/SS Elective[4]	4	0	0	4
					16

JUNIOR YEAR

Fall Semester — *Hour/Week*

CourseNo.	*CourseTitle*	*Class*	*Lab*	*Rec.*	*Cr*
CS 3314	*Design & Impl. Prog. Languages*	4	0	0	4
MA 2012	*Linear Algebra I (1/2semester)*	4	0	0	2
MA 2132	*Ordinary Diff. Equ. (1/2semester)*	4	0	0	2
	CS Elective[5]			4	
	HU/SS Elective[4]	4	0	0	4
					16

Spring Semester — *Hours/Week*

CourseNo.	*CourseTitle*	*Class*	*Lab*	*Rec.*	*Cr*
CS 3224	*Operating Systems*	4	0	0	4
CS 3414	*Design & Analysis Algorithms*	4	0	0	4
	CS Elective[5]			4	
	Sequence ElectiveI[6]				4
					16

SENIOR YEAR

Fall Semester — *Hour/Week*

CourseNo.	*CourseTitle*	*Class*	*Lb*	*Rec.*	*Cr*
CS 4513	*Software Engineering I*	2 1/2	1 1/2	0	3
PL 4062	*Computer Ethics*	2	0	0	2
	CS Elective[5]		3		
	HU/SS Elective[4]	4	0	0	4
	University Elective Sequence II[6]				4
					16

Spring Semester — *Hours/Week*

CourseNo.	*CourseTitle*	*Class*	*Lab*	*Rec.*	*Cr*
CS 4523	*Design Project II*	1	6	0	3
	CS Elective[5]		3		
	Technical Elective	4	0	0	3
	HU/SS Elective[4]	4	0	0	4
	Free Elective[8]				3
					16

1. Grade of C- or better is required in CS 1114, CS 1124 and CS 2134
2. Students who are placed by examination or by an adviser into EN 1080 must subsequently register for EN 1034, rather than EN 1014.
3. Students who are placed by examination or by an adviser into MA 914 must defer registration for MA 1024.
4. Approved HU/SS Electives have the following prefixes: AH, AN, EC, EN, HI, MU, PL and PS. Two courses must be from Level II Elective courses in different disciplines and one from Level III Advanced Elective courses.
5. At least two of the four required CS Electives must be chosen from the following: CS 392, CS 684, CS 308, CS 2204, CS 3254, CS 661. With departmental approval, certain graduate CS courses may also be used as CS Electives depending on course content and prerequisites.
6. A list of approved Sequence Electives is available from the Department of Computer and Information Science.
7. Approved Technical Electives courses for computer science majors can be in mathematics, management, industrial engineering, electrical and computer engineering and technical and professional communication provided they contain enough technical (or management) content that does not duplicate materials studied in other courses.
8. The Free Elective could be a course offered by any department, provided it does not duplicate material studied in other course.