1 General Information

1.1 College Station/Bryan

The College Station/Bryan area has a population of 135,000. Five of the six largest cities in Texas (Houston, Dallas, San Antonio, Ft. Worth, and Austin) are within 175 miles of the area. The Department has significant industrial connections in each of these cities. This is complemented by a growing industrial base in the local area. The area is known for a good quality of life that includes strong public school systems, parks, performing arts, sporting activities, and excellent, affordable housing.

1.2 Texas A&M University

Texas A&M University was established in 1876 as the first public institution of higher education in Texas. It was the first university in the nation to receive land-grant, sea-grant, and space-grant designations. It has 2,500 faculty; 7,400 graduate and professional students; and 36,000 undergraduate students on a spacious campus. Recent rankings of Texas A&M:

- Eleventh in research funding nationally.
- Fourth in new National Merit scholars, nationally.
- Sixth in total value of endowments, nationally.

1.3 The College of Engineering

The Dwight Look College of Engineering is one of the nation’s largest and is listed among the country’s top 20 graduate schools. It has 11 academic departments and is dedicated to providing a quality educational experience for its students. The college’s minority engineering, women’s, and honors programs are focal points in efforts to attract and retain students for engineering and science careers.

1.4 The Department of Computer Science

The computer science programs at Texas A&M began in 1962 and the department was formed in 1983. Continual change and growth has been a characteristic of the computing professions. This has been mirrored in the activities of the faculty. The Department includes 33 tenured and tenure-track faculty, 8 full-time lecturers, several part-time lecturers and more than 250 graduate students. The Department also offers the degrees of Bachelor of Science in Computer Science and in Computer Engineering to more than 1,100 undergraduate majors.

The Department moved into the Harvey R. Bright Building in 1990. The Department’s commitment to excellence reflects the University’s tradition of producing faculty and students whose leadership roles will help to ensure Texas’ major international position in the computing professions.

There is not a concise definition of Computer Science and Computer Engineering that is widely accepted. Our definition is reflected by our degree requirements. These requirements are changing as our profession changes. The current definition of our graduate programs are stated in section 4.

1.5 Graduate Program

Advanced study in computer science provides students with the skills to design and utilize modern computer systems. The Department encourages both fundamental research in computing and interdisciplinary research. This provides students with broad opportunities. Significant computer facilities and networks are provided to support student research. Research projects in diverse areas offer students a wide range of opportunities to gain experience while completing requirements for advanced degrees.

Graduate studies can lead to these degrees:

- Master of Science in Computer Science,
- Master of Science in Computer Engineering,
- Master of Computer Science,
- Master of Engineering in Computer Engineering,
- Doctor of Philosophy in Computer Science, and
- Doctor of Philosophy in Computer Engineering.

2 Admission

Admission to Texas A&M University and any of its sponsored programs is open to qualified individuals regardless of race, color, religion, sex, age, national origin, or educationally unrelated handicaps.

2.1 General Procedure for Applying

Applications for graduate study should be completed online: http://www.tamu.edu/admissions/. This online application is a “Common State of Texas” format. The applicant will specify the university, degree, semester of entry, etc. If this is impractical, contact the Department for a hardcopy application. Application fees, deadlines, required documentation, test scores, and other requirements are explained in the materials associated with either application format.
2.2 Academic Calendar

The academic year is composed of Fall, Spring, and Summer semesters. The Fall semester begins in late August and ends in early December. The Spring semester begins in mid-January and ends in early May. The Summer session begins in late May or early June and ends in early August. The Fall and Spring semesters are each 15 weeks. The Summer semester is 10 weeks in length and classes may be offered in one of two 5 week sessions. The Texas A&M University deadlines for international applications are: March 1 for the Fall and August 1 for the Spring. Fall admissions can start in the summer. Scholarship and fellowship decisions will likely be made on the body of applications received before January 2 for Fall applicants.

2.3 Entrance Requirements

Applicants must fulfill the requirements for admission to graduate studies as specified in the graduate catalog and should hold a bachelor’s degree in computer science, computer engineering, or a related field. Undergraduate preparation should include:

- Several high-level languages and software methodology,
- Data structures and analysis of algorithms,
- Digital logic, switching theory, and computer systems architecture,
- Operating systems, and
- Mathematics including calculus, linear algebra, probability and statistics, and discrete mathematics.

Students are responsible for all course prerequisites. They may use undergraduate courses to ensure said background. Students are encouraged to discuss their background with their committee chair and the graduate advisor.

All applicants are required to take the Graduate Record Examination (GRE) quantitative, verbal, and analytical tests. Applicants without a formal computer science/engineering degree should take the Computer Science subject test. The Test of English as a Foreign Language (TOEFL) is required of all foreign applicants.

The GRE performance of the applicants for the 2003–2004 academic year are shown in the following tables. The table on the left is a decades breakdown of the GRE components. The table on the right is a decades breakdown of the sum of the three standard GRE components. This is augmented with the percents of those in each decade that were admitted. This was based on more than 1,100 applications. The left table gives the minimum individual GRE scores for each decade of applicants. The right table shows the minimum sum of the GRE scores for these decades and the percent of successful applicants for each.

<table>
<thead>
<tr>
<th>Top %</th>
<th>min GRE</th>
<th>Top %</th>
<th>GRE sum</th>
<th>% Admit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td>A</td>
<td>V</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>800</td>
<td>800</td>
<td>700</td>
<td>10</td>
</tr>
<tr>
<td>20</td>
<td>800</td>
<td>780</td>
<td>660</td>
<td>20</td>
</tr>
<tr>
<td>30</td>
<td>800</td>
<td>760</td>
<td>640</td>
<td>30</td>
</tr>
<tr>
<td>40</td>
<td>790</td>
<td>750</td>
<td>610</td>
<td>40</td>
</tr>
<tr>
<td>50</td>
<td>790</td>
<td>730</td>
<td>580</td>
<td>50</td>
</tr>
<tr>
<td>60</td>
<td>780</td>
<td>700</td>
<td>560</td>
<td>60</td>
</tr>
<tr>
<td>70</td>
<td>760</td>
<td>680</td>
<td>530</td>
<td>70</td>
</tr>
</tbody>
</table>

All applications are reviewed by the Department’s Graduate Admissions and Awards Committee.

2.4 Evaluation Criteria

The criteria used in evaluation of applicants for admission to graduate study in computer science include:

- test scores,
- grade point average,
- relevant background,
- letters of reference,
- the applicant’s statement of purpose, and
- other relevant information developed by the department.

2.5 Your Application

We recommend that you include a brief resume (curriculum vitae) addressing university academics and closely related professional activities. Your Statement of Purpose and your resume should each be limited to one page with liberal margins, single spaced, and in a 10pt font. Information about secondary school performance, testing, and other pre-university studies is discouraged.

You should address academic topics in your Statement of Purpose that are verified by your university transcripts, references, or publications. You should not recall your earliest remembrances of exposure to a computer. References to your peer-reviewed publications in the open literature are helpful. Copies of informal publications are discouraged although URLs are welcomed.

If you are applying before graduation from your current degree program, indicate current and planned courses for which grades will not be on your transcripts.

You should note several items carefully. Be explicit that you are applying for admission to Computer Science or Computer Engineering. If you are applying to Computer Engineering, you also need to state which department, CPSC or ELEN.

If you state that you will seek a Ph.D., you will not seek a Masters degree along the way. If you state that you will seek a Masters degree, then all references to doctoral programs are distractions. If you are applying for a Masters program, be explicit that you are applying for a professional, terminal degree (MCS or MEN), see section 6.1.2, section 6.1.3, 6.2.2 or a research degree (MSCS or MSCE), see section 6.1.1, 6.2.1.

The preferred source of letters of recommendation is the faculty who taught your most advanced computing courses and are involved in computing research. The content of these letters that we are most interested in would include performance in the most advanced classes, rank in class, and scholarly aptitude. These letters should be mailed directly from the writer to this Department. We will accept them with the rest of your application if each letter is included in a sealed envelope with the writer’s signature across the seal. We will accept email letters in most cases.

Please include the email addresses and FAX numbers for: your references and academic department head.

Again, we are not interested in records from secondary school, technical schools, and certificate programs. You may include URLs that enable access to descriptions of your university programs but you are discouraged from including syllabi.
3 Financial Support

There are a number of teaching, non-teaching, and research assistantships available within the Department of Computer Science. These positions require 20 hours of work per week and typically have starting salaries of $1,100/month for Masters students and $1,500/month for Ph.D. students. The work assignments vary from classroom assistance to assisting with research. A small number of part-time assistant lecturer positions are available to suitable doctoral candidates.

Assistantships permit the holder to carry a full academic program of graduate work. Students with university financial support and health insurance qualify for resident tuition fees, see section 9.

Students applying for financial support should check the appropriate box(es) on the application form. Current students desiring financial support should file a GAT/GANT application form and should keep it current.

International students must pass the Texas A&M English Language Institute’s proficiency exam to qualify as a teaching assistant. This is often required for non-teaching assistantships.

Additional positions are available in other campus departments where students can work as programmers, systems analysts, and computer operators. Some of these positions may qualify for the same benefits (such as in-state tuition) as Departmental assistantships. Arrangements with other departments are made by the student.

Students may be employed by faculty as research assistants. These contacts may be initiated by individual faculty members based upon review of current students without assistantships and the list of students who have been admitted. Students are free to initiate these contacts. Most faculty do not want this contact initiated before the admission decision.

Ph.D. students receive the highest priority for Departmental funds (teaching assistantships, ...) followed by the research Masters degrees. Departmental needs also affect these priorities. Most faculty award their research assistantships similarly.

4 Program Requirements

Graduate students must fulfill the residence and scholastic requirements for graduate study as specified in the graduate catalog. In addition, the Department has these requirements:

Grade Point Requirements. Graduate students must maintain a grade point average of 3.0 in all coursework. A student whose grades drop below this level will be placed on probation and blocked from registration. A student on probation will be required to consult the graduate advisor. A student is usually given one semester to correct a grade deficiency.

Each graduate assistant funded by the Department must maintain a grade average of 3.25 in all coursework. A graduate assistant whose grades drop below this prescribed level will lose his/her assistantship.

Registration Requirements. University rules require continuous registration for all Fall and Spring semesters from the first semester of seeking a degree until its completion. Summer registration and the number of credits required in each semester are dependent upon funding, completion status, and (if applicable) visa status.

5 Facilities

The Department has extensive computer resources. Research laboratories are equipped with state-of-the-art workstations. Workstations, laser printers, and plotters are distributed throughout the Department in power-conditioned laboratories including approximately 500 workstations and a Hewlett-Packard V-class system. The University also provides additional resources.

6 Degree Requirements

Graduate students will be advised by the Departmental Graduate Advisor until they have formed an Advisory Committee and have an approved degree plan. Advisory Committee details are included with the description of each degree in this section. Appointments with the graduate advisor are made by calling 979-845-8981 or sending a request to: grad-advisor@cs.tamu.edu.

Each student is required to register for credit in CPSC 681 (Seminar) at least once per degree.

6.1 Computer Science

A research masters degree, a professional masters degree, and the Ph.D. are offered.

6.1.1 Master of Science

The Master of Science in Computer Science (MSCS) is a research degree, a thesis is required. The part of the final examination that is related to the presentation of the student’s research is public.

Advisory Committee Requirements. The student must select an Advisory Committee Chair from the Department’s graduate faculty. The chair, in consultation with the student and approval of the Department head, will select the remainder of the Advisory Committee. The student’s Advisory Committee for the Masters degree will consist of not fewer than three members. There must be at least one member from another department and there must be a majority from the Department. The student and chair will agree on a tentative Advisory Committee. The student will then personally interview each prospective committee member to determine whether the assignment will be accepted. The student, in consultation with the Chair and Advisory Committee, will then complete a degree plan. Each MSCS student must have formed an Advisory Committee and have an approved degree plan during the first semester after completion of 9 credits.

Degree Plan Requirements. The degree plan requirements for the MSCS degree include:

- The student must take at least 18 credit hours of graded CPSC graduate coursework (excluding 685).
- The student must pass, with a grade of B or better, three core courses, one selected from each of these sets: (627, 629), (613, 614) and (606, 608).
- At most one advanced undergraduate course.
6.1.2 Master of Computer Science — with Project

The Master of Computer Science should be thought of as a professional, terminal degree. Students pursuing this degree should not expect funding or to continue for a Ph.D. This degree plan is similar to the Master of Science except the formal thesis is replaced by a less formal report.

Advisory Committee Requirements. The Advisory Committee and approved degree plan must be completed in the first semester after completion of 9 credits. The process for forming the committee is identical to that outline in the Master of Science section, above.

Degree Plan Requirements. The degree plan requirements for the MCS degree include:

- The student must take at least 18 credit hours of CPSC graduate coursework (excluding 681 and 685).
- The student must pass, with a grade of B or better, three core courses, one selected from each of these sets: (627, 629), (613, 614) and (606, 608).
- At most one advanced undergraduate course.
- Graduate coursework taken outside the Department in supporting fields is limited to 3 credit hours of graduate courses.
- Three to five credit hours of CPSC 685 — Directed Studies. This is normally zero.
- The student must take 1 credit hour of 681 (seminar).
- Total of 30 credit hours (minimum.)

6.1.3 Master of Computer Science — Courses Only

The Master of Computer Science should be thought of as a professional, terminal degree. Students pursuing this degree should not expect funding or to continue for a Ph.D. This plan does not include a thesis, project, or final examination except for those in classes. This degree is expected to be offered remotely in the future.

Advisory Committee Requirements. Courses only MCS students will have a default advisory committee appointed by the Department. The Advisory Committee and approved degree plan must be completed in the first semester after completion of 9 credits.

Degree Plan Requirements. The degree plan requirements for the MCS degree include:

- The student must take at least 18 credit hours of CPSC graduate coursework (excluding 681 and 685).
- The student must pass, with a grade of B or better, three core courses, one selected from each of these sets: (627, 629), (613, 614) and (606, 608).
- At most one advanced undergraduate course.
- Graduate coursework taken outside the Department in supporting fields is limited to 3 credit hours of graduate courses.
- Three to five credit hours of CPSC 685 — Directed Studies. This is normally zero.
- The student must take 1 credit hour of 681 (seminar).
- Total of 30 credit hours (minimum.)

6.1.4 Ph.D. Degree

Advisory Committee Requirements. The Ph.D. advisory committee is like that of the Masters degrees except it must have a minimum of four members.

NOTE: Each Ph.D. student must have formed an Advisory Committee and have an approved degree plan by the end of their third semester.

Degree Plan Requirements. The degree plan for the doctoral program is flexible. The Ph.D. in computer science requires 96 credit hours after the bachelor’s degree. At most 32 hours credit for other graduate degree programs can be carried forward. Course distribution:

- The student must pass, with a grade of B or better, three core courses, one selected from each of these sets: (627, 629), (613, 614) and (606, 608).
- At most one advanced undergraduate course.
- Graduate coursework taken outside the Department in supporting fields is limited to 3 credit hours of graduate courses.
- Three to five credit hours of CPSC 685 — Directed Studies. This is normally zero.
- The student must take 1 credit hour of 681 (seminar).
- Total of 30 credit hours (minimum.)

Examination structure.

- Each Ph.D. student must pass the Preliminary Examination given by the Advisory Committee as described in the graduate catalog.
- Each Ph.D. student must pass the Final Examination (Dissertation Defense) given by the Advisory Committee as described in the graduate catalog. A final examination is required which includes a public presentation of the candidate’s research.

Dissertation requirements.

- A dissertation research proposal must be completed as described in the graduate catalog. The dissertation proposal may be presented at the preliminary exam at the discretion of the Advisory Committee.
The ability to perform independent research must be demonstrated by the dissertation. The dissertation must be the original work of the candidate. While acceptance of the dissertation is based primarily on its scholarly merit, it must also exhibit creditable literary workmanship. (Please see the graduate catalog for further details.)

The student is expected to submit a research paper to at least one refereed journal prior to the dissertation defense.

6.2 Computer Engineering

The Computer Engineering programs provide opportunities for students with interests in computer engineering to focus their studies more directly in this area. The following supplements the degree descriptions above and only covers the differences in requirements among the various degree programs.

NOTE: Computer Science faculty may co-chair computer engineering advisory committees along with a member of the Computer Engineering faculty.

General admission requirements are parallel to those stated in section 6.1 and the Texas A&M University Graduate Catalog.

In addition to these general requirements, the following specific requirements are expected to be met:

- Baccalaureate degree in engineering, physical science, mathematics, or related field.
- Students are required to have a background in the following areas: circuit theory, data structures, digital design, analysis of algorithms, electronics, software engineering, operating systems, computer architecture. These courses are described in the undergraduate catalog. Coursework taken to meet these prerequisites can not be counted for degree credit in the graduate programs. Students will be required to make up any deficiencies in these foundation courses.

6.2.1 Master of Science

This degree requires each student to write an acceptable thesis. The guidelines for thesis requirements and preparation as required by Texas A&M University are required by this program, and are described in the graduate catalog. A final oral examination in which the thesis work is presented is required. The part of the final examination that is related to the presentation of the student’s research is public.

Degree Plan Requirements.

- Graduate computer engineering courses from CPSC (at least 12 hours).
- Graduate computer engineering courses from ELEN that are not cross-listed with CPSC (at least 6 hours).
- Elective graduate courses prescribed by the student’s advisory committee, excluding 681 and 685 courses (11 hours).
- Seminar (CPSC 681). (1 hour).
- Project (CPSC 685). (4 or 5 hours).
- A total of at least 36 hours.

6.2.2 Master of Engineering — with Project

As of July 6, 2004 a 30 credit Master of Engineering in Computer Engineering degree (similar to the MCS) has not been approved. This degree requires one or two written reports to be submitted to the student’s advisory committee.

Degree Plan Requirements.

- Graduate courses from CPSC (at least 12 hours).
- Graduate courses from ELEN that are not cross-listed with CPSC (at least 6 hours).
- Elective graduate courses prescribed by the student’s advisory committee, excluding 681 and 685 courses (11 hours).
- Seminar (CPSC 681). (1 hour).
- Project (CPSC 685). (4 or 5 hours).
- A total of at least 36 hours.

6.2.3 Master of Engineering — Courses Only

This is not yet approved. This section will be reissued.

6.2.4 Doctor of Philosophy

Students who hold a related Masters degree may use some of their coursework to fulfill these requirements and must file a Ph.D. degree plan with at least 64 hours. If no Masters level degree is held by the student, a degree plan of at least 96 hours is required.

A preliminary exam is required and consists of written and oral exams. A written dissertation proposal is required and is often required to be presented orally. A dissertation is required. A final examination is required which includes a public presentation of the candidate’s research.

Degree Plan Requirements.

- These are being revised and this section will be reissued when these revisions are complete.
- Seminar (CPSC 681). (1 hour).
- Research (CPSC 691). (At least 32 hours).
- A total of at least 96 hours.

The examination structure and dissertation requirements are like those for the Ph.D. in Computer Science, see section 6.1.4.

7 Courses

This is a list of graduate courses offered by the Department of Computer Science. Related courses are also offered by the Department of Electrical Engineering.

All graduate courses meet 3 hours per week and carry three credits, unless otherwise noted. A listing of the courses annotated with appropriate faculty and a condensed catalog description follow. All courses assume the undergraduate degree as prerequisite. The instructor may also waive prerequisites.

601, 602, 603, 611, and 612. These courses are offered as service courses to the rest of the university community. Credit in these courses may be used to satisfy prerequisites but not toward a graduate degree in Computer Science or Computer Engineering.
605 Compiler Design. Chen, Friesen, Furuta, Rauchwerger, Volz. Advanced topics in compiler writing; parser generators and compiler-compilers; dynamic storage and scope resolution; data flow analysis and code optimization. Prerequisite: CPSC 434.

606 Software Engineering. Childs, Lively, Simmons, Williams. Development of advanced concepts in software engineering; software development environments as a mechanism for enhancing productivity and software quality; the classification, evaluation, and selection of methodologies for environments; rapid prototyping and re-usability concepts; artificial intelligence techniques applied to software engineering; transformational systems; maintenance environments. Prerequisite: CPSC 431.

607 Software Models and Metrics. Lively, Simmons. Software models and metrics; productivity predicting techniques; complexity measures; software reliability models; cost estimation models. Prerequisite: CPSC 431.

608 Database Systems. Li, Leggett. Basic database models; concepts in database systems including knowledge representation and knowledge-based systems; database architectures, network and distributed databases; database design and performance measurements; query languages and natural language interfaces. Prerequisite: CPSC 310.

609 Artificial Intelligence Approaches to Software Engineering. Lively, Simmons, Williams. Artificial Intelligence techniques and approaches to software engineering; revolutionary paradigms and automatic programming; the knowledge-based assistant in management, specification capture, prototyping and maintenance; transformational systems and reusable libraries; AI tools for software development; graphical approaches to software engineering. Prerequisites: CPSC 606 and 625.

610 Hypertext/Hypermedia Systems. Furuta, Kerne, Leggett, Shipman. Comprehensive coverage of the area of hypertext/hypermedia systems. Course content: the history of hypertext, a survey of current hypertext systems, and research directions in hypertext. Appropriate database models, information retrieval models and user interface models are studied.

611 Operating Systems. Pooch, Zhao. Analysis of algorithms in operating systems; sequencing and control algorithms supporting concurrent processes; scheduling algorithms to minimize execution time and mean flow times; algorithms for allocating tasks to processors.

612 Computer Architecture. Kim, Liu, Lively, Mahapatra, Pooch, Rauchwerger, Simmons, Walker, von Neumann architecture and limitations; concepts of parallel computer structures and concurrent computation; analysis of pipeline computers, array processor machines and multiprocessor systems; control models for control-driven, data-driven and demand-driven machines. Discussion and analysis of intelligent interfaces, inference machines and knowledge-based systems. Prerequisite: CPSC 410.

613 Operating Systems. Liu, Loguinov, Mahapatra, Pooch, Zhao. Computer network concepts including network architecture, layering, protocols, packet switching and virtual circuits; performance evaluation and design considerations for local area networks; packet distributed networks; satellite networks; distributive processing including array, parallel and multiprocessor systems. Prerequisite: CPSC 463.

614 Computer Architecture. Kim, Liu, Lively, Mahapatra, Pooch, Rauchwerger, Simmons, Walker, von Neumann architecture and limitations; concepts of parallel computer structures and concurrent computation; analysis of pipeline computers, array processor machines and multiprocessor systems; control models for control-driven, data-driven and demand-driven machines. Discussion and analysis of intelligent interfaces, inference machines and knowledge-based systems. Prerequisite: CPSC 410.

615 Distributed Component Architecture. Introduce general techniques and approaches of software architecture (e.g. architecture style, ADL, ACME, UML, DSSA, distributed component and middleware); software life cycles; investigate distributed component architectures, (e.g. CORBA, COM/DCOM, JavaBeans) as specific examples of architecture for in-depth knowledge. Prerequisite: knowledge of at least one object oriented language.

616 Resilient Computer Systems. Liu, Mercer (ELEN), Pooch, Walker. Basic design of fault-tolerant computer systems, models of reliability and availability in fault-tolerant systems; system reorganization and self-repair; analysis of systems capable of non-stop processing in response to environmental changes, varying functional requirements and user needs. Prerequisite: CPSC 410. Cross-listed with ELEN 618.

617 Networks and Distributed Computing. Liu, Loguinov, Mahapatra, Pooch, Zhao. Computer network concepts including network architecture, layering, protocols, packet switching and virtual circuits; performance evaluation and design considerations for local area networks; packet distributed networks; satellite networks; distributive processing including array, parallel and multiprocessor systems. Prerequisite: CPSC 463.


619 Parallel Geometric Computing. Amato, Chen, Friesen. Parallel computer architectures and algorithms for solving geometric problems raised in VLSI design, pattern recognition and graphics; advanced research results in computational geometry including convexity, proximity, intersection, geometric searching and optimization problems. Prerequisites: CPSC 311 or ELEN 350. Cross-listed with ELEN 623.


621 Parallel Algorithm Design and Analysis. Amato, Chen, Friesen, Rauchwerger, Welch. Design of algorithms for use on highly parallel machines; area-time complexity of problems and general lower bound theory; application (of these concepts) to artificial intelligence, computer vision and VLSI design automation. Prerequisite: CPSC 629.

622 Theory of Computability. Chen, Friesen, Sze, Welch. Formal models of computation such as pushdown automata, Turing machines and recursive functions; unsolvability results; complexity of solvable results. Prerequisites: CPSC 433.
629 Analysis of Algorithms. Amato, Chen, Friesen, Welch. Concrete algorithm design and analysis; abstract models to analyze the complexity of problems; NP-completeness; approximation and probabilistic algorithms. Prerequisite: CPSC 311.

631 Programming Environments for Artificial Intelligence. Ioerger, McCormick. Advanced techniques for knowledge-based systems. AI planning systems: linear and nonlinear planning, reactive planning, constraint-based planning, and meta-planning; qualitative reasoning: qualitative models, qualitative simulation, temporal and spatial reasoning; advanced models for reasoning under uncertainty: influence diagrams, Dempster-Shafer reasoning, and fuzzy logic; advanced cognitive architectures; truth maintenance systems; and other topics in knowledge engineering and processing. Prerequisites: CPSC 625.

632 Expert Systems. Ioerger, McCormick, Simmons, Williams. Basic concepts for building expert systems; inference strategies; applications and case studies; techniques for knowledge acquisition; use of existing tools for building expert systems. Prerequisite: CPSC 625.

633 Machine Learning. Gutierrez-Osuna, Ioerger, Kerne. This course will survey machine learning techniques, which include induction from examples, conceptual clustering, explanation-based learning, exemplar learning and analogy, discovery, and genetic algorithms. Prerequisite: CPSC 625.

634 Intelligent User Interfaces. Shipman. Intersection of artificial intelligence and computer-human interaction: emphasis on designing and evaluating systems that learn about and adapt to their users, tasks, and environments.


637 Complexity Theory. Chen, Friesen, Welch. Deterministic, non-deterministic, alternating and probabilistic computations; reducibilities; P, NP, and other complexity classes; abstract complexity; time, space and parallel complexity; and relativized computation. Prerequisites: CPSC 627.

639 Fuzzy Logic and Intelligent Systems. Langari (MEEN). This course covers concepts and techniques in fuzzy logic as well as their applications for developing intelligent systems for control, decision making, and pattern recognition. Topics to be covered include fuzzy sets, fuzzy rule-based inference, fuzzy logic control, possibility theory and its relationship to probability theory, fuzzy expert systems, neuro-fuzzy systems, and fuzzy pattern recognition. Prerequisite: CPSC 320 or 625.


644 Cortical Networks. Choe, McCormick. Architecture of the mammalian cerebral cortex; its modular organization and its network for distributed and parallel processing; cortical networks in perception and memory; neuronal microstructure and dynamical simulation of cortical networks; the cortical network as a proven paradigm for the design of cognitive machines. Prerequisites: CPSC 625.


646 The Digital Image. Akleman (VIZA), Keyser, House (VIZA). Tools and techniques for generation, handling, and analysis of two dimensional digital images; image representation and storage; display, media conversion, painting and drawing; warping; color space operations, enhancement, filtering and manipulation. Prerequisite: VIZA 653. Cross-listed with VIZA 654.

647 Image Synthesis. Akleman (VIZA), Keyser, House (VIZA). Principles of image synthesis from 3-D scene descriptions; includes local and global illumination, shading, shadow determination, hidden surface elimination, texturing, raster graphics algorithms, transformations and projects. Prerequisites: VIZA 653. Cross-listed with VIZA 656.

648 Computer Aided Sculpting. Akleman (VIZA), Keyser, House (VIZA). Mathematical and artistic principles of 3-D modeling and sculpting; includes proportions, skeletal foundation, expression and posture, line of action; curves, surfaces and volumes, interpolation and approximation, parametric and rational parametric polynomials, constructive solid geometry, and implicit representations. Cross-listed with VIZA 657.

649 Physically-Based Modeling. Akleman (VIZA), Keyser, House (VIZA). Physical simulation as used in choreography, geometric modeling, and the creation of special effects in computer graphics; a variety of problems and techniques explored which may include particle-methods, modeling and simulation of flexible materials, kinematics and constraint systems. Cross-listed with VIZA 659.

651 Simulation I. Childs, Pooch, Williams. Introduction to simulation and comparison with other problem solving techniques; simulation methodology including generation of random numbers and variates, time flow mechanisms, sampling considerations, and validation and analysis of simulation models and results; survey of discrete simulation languages; applications of simulation, including operating systems and networks. Prerequisite: Knowledge of several programming languages.


661 Integrated Systems Design Automation. Walker. Design verification and optimization, high level synthesis, incremental synthesis; simulation of IC and systems testing and self-testing; probabilistic methods, spectral techniques; wafer-scale integration. Prerequisite: CPSC 442.

662 Distributed Processing Systems. Bettati, Loguinov, Pooch, Welch, Zhao. Principles and practices of distributed processing; protocols, remote procedure calls; file sharing; reliable system design; load balancing; distributed database systems; protection and security; implementation. Prerequisite: CPSC 463.

663 Real-Time Systems. Bettati, Liu, Pooch, Zhao. Taxonomy of real-time computer systems; scheduling algorithms for static and dynamic real-time tasks; hard real-time communications protocols; programming languages and environments for real-time systems; case studies of real-time operating systems. Prerequisites: CPSC 456.


668 Distributed Algorithms and Systems. Friesen, Welch, Zhao. Study of algorithms for distributed computer systems, especially loosely-coupled and failure-prone systems; formal models, algorithm design and analysis, lower bounds and impossibility proofs. Prerequisite: CPSC 629.

669 Computational Optimization. Chen, Friesen. Combinatorial theory of polytopes as a tool for the solution of combinatorial optimization problems; applications to max flow, matching and matroids; geometric interpretation of the results indicating the profound role that polyhedral combinatorics plays in the design and complexity of approximation algorithms. Prerequisite: CPSC 629.

670 Information Storage and Retrieval. Furuta, Leggett. Information retrieval deals with the representation, storage, and access to very large multimedia document collections. Fundamental data structures and algorithms of current information storage and retrieval systems and relates various techniques for the design and evaluation of complete retrieval systems. Algorithms for indexing, compressing, and querying large digital collections are studied.

671 Computer-Human Interaction. Furuta, Kerne, Leggett, Lively, Shipman. Course content includes the history and importance of computer-human interaction (CHI), theories of CHI design, modeling of computer users and interfaces, empirical techniques for task analysis and interface design, styles of interaction and future directions of CHI.

672 Computer-Supported Collaborative Work. Furuta, Leggett, Li, Shipman. Design, implementation and use of technical systems that support people working cooperatively; current theoretical, practical and social issues in CSCW and future directions of the field; theoretical models of cooperative work, computer-mediated communication, group decision support systems, situation theory, and technical innovations such as electronic meeting rooms, liveboards, shared editors, and synchronous and asynchronous communication technologies. Prerequisite: CPSC 610 or 671.

673 Information, Secrecy, and Authentication I. Pooch, Blakley (MATH). Preliminaries: probability, information, entropy, signals, channels; group-theoretic view of messages; contemporary secrecy and digital signature systems; one-time pads, DES, RSA, DSS, wheels, LFSR-based systems; analog scramblers; key exchange, key management, secret sharing, access structures; measures of security. Cross-listed with MATH 673.

674 Information, Secrecy, and Authentication II. Pooch, Blakley (MATH). Classical and recent attacks; login, compression, error control, and genetic codes; finite and infinite codes; matrices, graphs, duals, groups, morphisms, composites, products, rates, and classification of codes; the confusion/diffusion/arithmetic/calculus extension of Shannon's two design primitives. Prerequisites: MATH 634. Cross-listed with MATH 674.

675 Digital Libraries. Furuta, Leggett. Surveys current research and practice in Digital Libraries, which seek to provide intellectual access to large-scale, distributed digital information repositories; current readings from the research literature which covers the breadth of this interdisciplinary area of study.
8 Graduate Faculty/Research Staff

Nancy Amato, Associate Professor. B.S. in Mathematical Sciences, A.B. in Economics (1986), Stanford University; M.S. in Computer Science (1988), University of California, Berkeley; Ph.D. in Computer Science (1995), University of Illinois at Urbana-Champaign. Areas of Interest: motion planning, robotics, computational geometry, CAD, virtual reality, parallel and distributed computing, parallel algorithms, performance modeling.

Riccardo Bettati, Associate Professor. Diploma in Computer Science (1988), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland; Ph.D. in Computer Science (1994), University of Illinois. Areas of Interest: very large distributed real-time systems, real-time communication, distributed multimedia, intelligent networks.

Jianer Chen, Professor. B.S. in Computer Science (1982), Central-South University of Technology, China; M.S. in Computer Science (1984), New York University; M.A. in Mathematics (1989), M.Phil. in Mathematics (1990), Columbia University, NY; Ph.D. in Computer Science (1987), New York University; Ph.D. in Mathematics (1990), Columbia University. Areas of Interest: complexity theory, algorithm analysis, parallel processing, and combinatorics.

S. Bart Childs, Professor and Graduate Advisor. B.S. in Civil Engineering (1959), M.S. in Civil Engineering (1960), Ph.D. (1966), Oklahoma State University. Areas of Interest: computational science and engineering, literate programming, documentation, software engineering, and programming environments.

Yoonsuck Choe, Assistant Professor. B.S. in Computer Science (1993), Yonsei University (Seoul, Korea); M.A. (1995) and Ph.D. (2001) in Computer Science, University of Texas at Austin. Areas of Interest: neural networks, computational neuroscience, human visual perception, computer vision, and artificial intelligence.

Donald K. Friesen, Professor and Associate Head. B.A. in Mathematics (1963), Knox College, IL; M.A. in Mathematics (1965), Ph.D. in Mathematics (1966), Dartmouth College; Ph.D. in Computer Science (1978), University of Illinois. Areas of Interest: algorithm analysis, parallel algorithms, and artificial intelligence.


Thomas Loerger, Associate Professor. B.S. in Molecular and Cell Biology (1989), Pennsylvania State University; M.S. in Computer Science (1992), Ph.D. in Computer Science (1996), University of Illinois. Areas of Interest: artificial intelligence, machine learning, and bioinformatics.


Andreas Klappenecker, Assistant Professor. Diplom in Computer Science (1995), University of Karlsruhe; Ph.D. in Computer Science (1998), University of Karlsruhe. Areas of Interest: Quantum computing, cryptography, and wavelets.

John J. Leggett, Professor and Associate Head. B.B.A. in Computer Science (1974), Angelo State University; M.C.S. in Computer Science (1976), Ph.D. in Computer Science(1982), Texas A&M University. Areas of Interest: digital library systems, hypermedia systems, collaborative systems, computer-human interaction, and operating systems.

Du Li, Assistant Professor. B.S. in Computer Science (1992), Wuhan University; M.S. in Computer Science (1995), Peking University; Ph.D. in Computer Science (2000), UCLA. Areas of Interest: CSCW, Internet and distributed computing, logic programming, programming languages, and database systems.
Frank M. Shipman III, Associate Professor. B.S.E.E. in Electrical Engineering (1988), Rice University; M.S. in Computer Science (1990), Ph.D. in Computer Science (1993), University of Colorado. Areas of Interest: real-time distributed computing systems, high speed networking and wavelet applications.


Bruce H. McCormick, Professor. B.S. (1950), Massachusetts Institute of Technology; Ph.D. (1955), Harvard University. Areas of Interest: scientific visualization and modeling, computer vision, neural networks, and artificial intelligence.

Paul Nelson, Professor emeritus. B.S. (1958), Auburn University; M.S. (1962), Ph.D. (1969), University of New Mexico. Areas of Interest: mathematical software, numerical analysis, and parallel numerical analysis.

Udo W. Pooch, Professor, E-Systems Professor. B.S. (1963), University of California, Los Angeles; Ph.D. (1969), University of Notre Dame. Areas of Interest: operating systems, system architecture, computer networking, fault-tolerant systems, and real-time computing.


Vivek Sarin, Assistant Professor. B.Tech. in Computer Science and Engr. (1990), IIT Delhi; M.S. in Computer Science (1993), Univ. of Minnesota; Ph.D. in Computer Science (1997), University of Illinois. Areas of Interest: numerical algorithms, scientific computing, parallel and distributed computing.

Frank M. Shipman III, Associate Professor. B.S.E.E. in Electrical Engineering (1988), Rice University; M.S. in Computer Science (1990), Ph.D. in Computer Science (1993), University of Colorado. Areas of Interest: intelligent user interfaces, hypertext, computer-supported cooperative work, and computer-human interaction.

Dick B. Simmons, Professor. B.S. in Electrical Engineering (1959), Texas A&M University; M.S. in Electrical Engineering (1961), Ph.D. in Computer and Information Sciences (1968), University of Pennsylvania. Areas of Interest: artificial intelligence, software engineering, computer architecture, and expert systems.


Jennifer L. Welch, Professor. B.A. in Liberal Arts (1979), University of Texas at Austin; S.M. in Computer Science (1984), Ph.D. in Computer Science (1988), Massachusetts Institute of Technology. Areas of Interest: distributed and parallel algorithms, distributed computing, and fault tolerance.


Wei Zhao, Professor and Associate Vice President for Research. DIPC (1977), Shaanxi Normal University, China; M.S. (1983), Ph.D. (1986), University of Massachusetts. Areas of Interest: real-time computing, distributed operating systems, and computer networks.
9  More Information

Departmental Information
WWW: http://www.tamu.edu
Department of Computer Science
Texas A&M University
College Station, TX USA 77843-3112

Office of Admissions
Texas A&M University
College Station, TX USA 77843-0100
Telephone: 979-845-1004
FAX: 979-845-0727
Email: international-admission@tamu.edu
Email: graduate-admission@tamu.edu

International Student Services
Texas A&M University
College Station, TX USA 77843-1226
Telephone: 979-845-1824
FAX: 979-862-4633
Email: iss@tamu.edu

Office of Graduate Studies
Texas A&M University
College Station, TX USA 77843-1113
Telephone: 979-845-3631
FAX: 979-845-1596
Email: ogs@tamu.edu

Graduate Advisor
Department of Computer Science
Texas A&M University
College Station, TX USA 77843-3112
Telephone: 979-845-8981
FAX: 979-862-3684
Email: grad-advisor@cs.tamu.edu
WWW: http://www.cs.tamu.edu

CS Graduate Student Organization
Email: csgsa@cs.tamu.edu

On-line fee information.
WWW: http://www-fiscal.tamu.edu/sfs/