The School of Information and Computer Science

Established in 1963, the School of Information and Computer Science is rapidly improving its position as one of the better computer science departments in the country. A 1982 survey by the Conference Board of Associated Research Councils ranked the School as "most improved" among the fifty-seven Ph.D.-granting computer science departments in the country, and the school has maintained this position. The School's 1985 graduate program is ranked among the top five in the nation, and the school has been rated among the top ten in the nation in terms of faculty research productivity.

The educational programs of the School are supported by a wide variety of computer equipment. Both introductory and advanced classes use the IBM Computer Supported Instruction Laboratory, which includes two classrooms, each with thirty IBM PCs connected to an IBM 4361 through three IBM Series 1 systems. Another classroom/laboratory contains ten AT&T 382 systems. Special equipment for instruction in computer graphics includes a Tektronix 4130 system, twelve Tektronix 4107 graphics terminals, and four color inkjet printers. Some IBM classes use the facilities of the campus computing center, which operates four CDC CYBER 150 systems, 8010, 8300, and 9000, an IBM 4381, two AT&T 3580s, fifteen AT&T 3550 dot-mapped terminals, and a Xerox 7900 laser printer.

Research

Computer science faculty members perform research in areas that span the fields of information and computer science. Faculty in many of these areas are nationally and internationally recognized experts. Many faculty members contribute to the research environment in the School of Information and Computer Science.

Other research includes a cluster of these VAX 11/780 systems, a Data General MV/10000, and an Amdahl 470D computer system. The School owns a high-speed printer, used for high-quality printing of technical papers and research proposals. Many of the School's computer systems are networked with the National Science foundation, which make it possible to connect to a campus-wide network. The latter network provides dial-in access to any machine on the network. The School also owns a number of specialized research equipment, including Symbolics 3600 Lisp machines, used for research in artificial intelligence.

In addition to supporting the educational programs of the School, the campus computing center operates a Pyramin 90X system that is used by the School for research purposes. The Pyramin allows the use of both Unix and Macintosh systems.
What's New in ICS

During 1985, Georgia Tech will celebrate its Centennial. As Tech prepares to begin its second century, the School of Information and Computer Science is playing an increasingly important role in the Institute, as the following highlights of 1984-85 show.

New computer science building is planned

Funds have been allocated and an architectural firm chosen for the planning of a new building for computer science and computer engineering. The planned building would house all ICS offices and laboratories. In the meantime, the School's acute need for space will be eased by the gain of additional office space in the adjacent Himmann building in the fall of 1985. The new offices will house the Software Test and Evaluation Project and the artificial intelligence research group.

Herbert Simon visits Tech

Dr. Herbert A. Simon of Carnegie-Mellon University visited Tech in May, 1985, and presented a Centennial Lecture titled "Understanding Intelligence: Human and Artificial" to a crowd of about 500 faculty, staff, and students. Later the same day, Professor Simon visited the School of ICS and met with faculty and students. Simon, a Nobel Prize winner, is one of the pioneers of artificial intelligence.

Ph.D. enrollment grows

Ph.D. enrollment in ICS has increased dramatically over the past few years. The following table shows Fall Quarter enrollment in the Ph.D. program for the past four-year period.

<table>
<thead>
<tr>
<th>Year</th>
<th>Fall 1982</th>
<th>Fall 1983</th>
<th>Fall 1984</th>
<th>Fall 1985</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>131</td>
<td>140</td>
<td>161</td>
<td>195</td>
</tr>
</tbody>
</table>

One major change in the Ph.D. program took effect during the 1984-85 academic year: students are no longer required to satisfy a foreign language requirement.

Evening M.S. program to begin

The School will again offer its popular evening M.S. program beginning Spring Quarter 1986. The program allows students to earn their M.S. degree in nine quarters by taking two courses per quarter at night.

New faculty join ICS

The School was fortunate to hire four new faculty members in 1985:

- Mustapha Ahmad, who will finish his Ph.D. work at the State University of New York at Stony Brook in 1986, will join the School as an assistant professor. Professor Ahmad specializes in distributed operating systems and languages, and his dissertation is titled "Multilevel Communication in Distributed Algorithms."
- Mostafa H. Ammar will also join the School as an assistant professor. He will complete his dissertation on the modeling and performance evaluation of computer and telecommunications systems using simulation and queuing theory.
- Richard E. Callaghan will join the School at the rank of professor. He received the Ph.D. from Yale University in 1977 and was previously an associate professor at the University of Connecticut. He specializes in artificial intelligence, particularly natural language processing, knowledge representation, and problem solving.
- Gary L. Peterson will join the School as an associate professor. He received the Ph.D. from the University of Washington in 1979 and was previously an assistant professor at the University of Rochester. Professor Peterson specializes in theoretical computer science, particularly parallel computation, concurrent control, parallel networks, and theory of VLSI layouts, complexity theory, and algorithms.

The School will also host three visiting faculty members during the 1985-86 academic year:

- Tomasz Kowalowski will hold the rank of visiting associate professor. He will be on leave from the University of Wroclaw, Poland. Professor Kowalowski received his Ph.D. from the University of California at Berkeley in 1973. He specializes in programming language design and implementation.
- Bruce Lownie, who will hold the rank of visiting professor, is visiting from the IBM Thomas J. Watson Research Center in Yorktown Heights, N.Y. Professor Lownie received the Ph.D. from the University of Michigan in 1966 and specializes in programming languages, software engineering, and software development tools.

New equipment arrives

In 1985, the School received eighteen AT&T 3B22 systems, eight AT&T 4425 terminals, and eight AT&T 5620 dot-mapped terminals as part of a $2.5 million AT&T donation to Georgia Tech. The School also received a donation of an MVS/10000 computer system from Data General Corporation. Software for the MV/10000 includes Data General's AIX/RT/OS, an Ada development environment, and a UNIX host environment.

Purchases by the School include two additional Symbolics 3600 SLE/PMs for research in artificial intelligence and an Imagen laser printer for high-quality printing of technical papers and research proposals.

The campus computer center upgraded its two DEC CYBER 170 systems and purchased two other CYBERs, giving Tech a total of four CYBER 160 systems (models 810, 3000, 355, and 500). The computer center also upgraded its IBM 4341 to a 4391 and received equipment from the AT&T donation: two AT&T 3B22 systems, a new 6924 terminal, an AT&T 3B22 terminal, and thirty AT&T 5620 dot-mapped terminals.

New special program in computer networking and communications systems continues to develop

During 1984-85, the School's new Special Program in Computer Networking and Communications Systems received donations of hardware and software from Atlantic Research, Bell Northern Research, Bell-South International, Connexions Inc., Dynascan, Gandalf, Idacom, Intelligent Systems, Paradyne, Teknek, and Telecommunications Technics Corporation. With these donations, the program has received support totaling $64 million from 30 companies.

Three new lecture/laboratory courses in networking and communications are now offered on a regular basis. These courses make use of the computer network and communications systems laboratory, which is now fully operational.

Raymond Miller

Director
Faculty

Musique Ahram, Assistant Professor, Ph.D., State University of New York at Stony Brook, 1989. Distributed operating systems and languages, distributed algorithms, communication protocols.

Mostafa H. Ammar, Assistant Professor, Ph.D., University of Waterloo, 1989. Communication networks and protocols, large scale information delivery systems, queuing theory.

Albert H. Sade, Associate Professor, Ph.D., University of Michigan, 1973. Human-computer interaction, adaptive computer systems, human factors in computer systems.

W. Gum Bai, Research Scientist, M.S., Georgia Institute of Technology, 1990. Applications systems, real-time software, computer-supported education.

P.-Y. Chen, Assistant Professor, Ph.D., University of Illinois at Urbana-Champaign, 1981. Computer system architecture, computer networks.

Lucia Chiricosta, Professor and Associate Director, Ph.D., Emory University, 1981. Computer science education.

Richard C. Ewell, Professor, Ph.D., Yale University, 1977. Natural language processing, commonsense reasoning and problem solving, knowledge representation, architectures for intelligent systems, cognitive modeling.

P. D. Dasgupta, Assistant Professor, Ph.D., State University of New York at Stony Brook, 1984. Distributed operating systems, distributed database systems, concurrency control, fault-tolerance, networks.

Richard A. DeMillo, Professor and Assistant Director for Research, Ph.D., Georgia Institute of Technology, 1972. Software engineering, VLSI systems, computer security, theoretical computer science.

Philip K. Eustace, Jr., Professor, Ph.D., Stanford University, 1965. Distributed systems, computer networks, telecommunications systems, computer systems, data communications, operating systems.

John J. Godd, Jr., Assistant Professor, M.S., University of Massachusetts, 1969. Computer programming, parallel processing.

Mark H. Grabrow, Assistant Professor, Ph.D., University of Toronto, 1981. Database, logic, theoretical computer science.


Oscar H. B. Grob, Assistant Professor, Dr., Massachusetts Institute of Technology, 1981. Local area networks, modeling and analysis of computer systems, computer networks.


Francis C. Kaiser, Associate Professor and Librarian, M.A., Emory University, 1963. Librarianship, information retrieval, indexing, literature search, documentation.

K. N. King, Assistant Professor, Ph.D., University of California at Berkeley, 1980. Programming languages, theoretical computer science.

Janet L. Koehler, Associate Professor, Ph.D., University of Virginia, 1980. Artificial intelligence, cognitive science, models of long-term memory, the role of experience in learning.

Richard J. Litch, Jr., Associate Professor, Ph.D., University of Wisconsin at Madison, 1972. Programming languages and environments, compilers, distributed processing.

Martin D. McAndrew, Assistant Professor, Ph.D., University of Illinois at Urbana-Champaign, 1981. Distributed systems, operating systems, real-time systems, on-leave, 1986.

Raymond E. Miller, Professor and Director, Ph.D., University of Illinois at Urbana-Champaign, 1967. Theory of computation, machine organization, parallel processing, computer communication, programming languages.

Bruce F. Nasher, Assistant Professor, Ph.D., University of Texas at Dallas, 1981. Computer graphics, geometric modeling, VLSI.

John F. Passaline, Jr., USAF, Senior Research Engineer and Assistant Director for Laboratories, M.S.E., Purdue University, 1963. Software engineering, computer architecture.

Gary L. Peterson, Associate Professor, Ph.D., University of Washington, 1979. Theoretical computer science, parallel computation, concurrent control, network theory, distributed systems, complexity theory, algorithms.

J. N. Poon, Associate Professor, Assistant Vice President for Academic Affairs and Assistant to the President for Information Technology, Ph.D., Georgia Institute of Technology, 1970. Computer-based education.

Philip J. Siegelman, Associate Professor, Ph.D., Ohio State University, 1980. Evaluation and statistical techniques applied to information retrieval systems, documentation, representation methodology, applications of measures of information to studies.


Gopalakrishnan Vajaj, Assistant Professor, Ph.D., Princeton University, 1981. VLSI design tools, algorithms and complexity theory, graph theory, combinatorics, programming languages.

Pranav Zunle, Professor, Ph.D., Georgia Institute of Technology, 1968. Information sciences, information processing systems, systems theory, implications of computer recognition, computer integrated manufacturing systems.

Visitors

Tomasz Kowalski, Visiting Associate Professor, Ph.D., University of California at Berkeley, 1973. Programming language design and implementation. (Visiting from the University of California at Berkeley, 1973)

Burt Learson, Visiting Professor, M.S., University of California, 1973. Programming languages, software engineering, software development tools. (Visiting from IBM Thomas J. Watson Research Center, Yorktown Heights, N.Y.)

Aditya P. Mathur, Visiting Associate Professor, Ph.D., Berkeley University, 1987. Software engineering, computer architecture.


J. C. Smith, Visiting Scholar, Ph.D., School of Engineering, 1984. Database systems, queuing theory, data communications, operating systems. (Visiting from MIT, 1984)


Karen Hutchenson, Sharon Chamberlain, Debra Woods, Ann Lewis

Adminiistrative and Secretarial

Edmund R. Rameau, Assistant to the Director

Robert L. Bullock, Administrative Assistant

Sharon Chamberlain, Administrative Secretary

Mavis Cannon, Senior Secretary

J. T. C. Cook, Word Processor Operator

Karen Hutchenson, Administrative Secretary

Ann H. Lewis, Administrative Secretary

Esther E. Richards, Administrative Secretary

Ann K. Richtsmeier, Senior Secretary

Debra Woods, Staff Assistant

Technical

Jeannine B. Baskin, Research Scientist

Ahmed N. Bagli, Senior Research Scientist

Walter E. Coleman, Operations Manager, IS/IS/IS

Jeffrey L. Grover, Research Scientist

Ronald R. Mathieu, Research Scientist

Stephen M. Johnson, Software Specialist

Jeffrey E. Lee, Software Specialist

Arnold D. Robbins, Software Specialist

William G. Putnam, Research Scientist

Michael McCracken, Senior Research Engineer

B. B. Mathur, Visiting Scholar, Ph.D., School of Engineering, 1984. Database systems, queuing theory, data communications, operating systems. (Visiting from MIT, 1984)

Jeffrey E. Lee, Software Specialist

Win Strickland, Jr., Software Specialist

Charles L. Hall, Director of Engineering, 1984. Database systems, queuing theory, data communications, operating systems.

Win Strickland, Jr., Software Specialist

Karen Hutchenson, Sharon Chamberlain, Debra Woods, Ann Lewis

Part-Time Faculty

Leone J. Buck, Instructor in Electronics, Brookings Systems

Daniel H. Forgy, Jr., Instructor in Medical Systems Development Corporation

Ross Pagano, Associate Professor, Georgia Tech Research Institute

Lawrence J. Gallo, Associate Professor, Georgia Tech Research Institute

M. David Price, Professor, Lockheed Georgia Company

Robert M. Siegelman, Associate Professor, Southern Bell Telephone Company

Charles L. H., Instructor in E.S.A/C/COM Corporation

Karen Hutchenson, Sharon Chamberlain, Debra Woods, Ann Lewis

Georgia Tech

The Georgia Institute of Technology, founded in 1885, offers educational programs in science, engineering, management, and architecture. Georgia Tech is a unit of the University System of Georgia, which operates thirty-three colleges and universities. Tech currently enrolls 11,000 students, representing every state and eighty countries. The quality of the student body is high; Tech enrolls the highest percentage of Phi Beta Kappa National Merit Scholars and National Achievement Scholars among publicly-supported U.S. colleges.

Athletic programs at Georgia Tech are nationally famous, as are the "Ramblin' Wreck" fans and other school traditions. The football, basketball, and baseball teams are all ranked in the top twenty nationally at some time during the 1980-85 season. The basketball team was particularly successful, winning the championship of the Atlantic Coast Conference and finishing the season ranked sixth in the nation. The Arthur R. Edgeworth Athletic Center, a $6.5 million facility that opened in early 1982, is a symbol of Tech's commitment to intercollegiate athletics.

The campus of Georgia Tech covers three hundred acres near downtown Atlanta. The School of Information and Computer Science occupies a portion of the Rich Building, a modern structure near the center of campus that also houses the Office of Computing Services.

Located near the Rich Building, the Prudential Memorial Library contains over 1.7 million volumes and 2.3 million microforms and receives 26,000 periodicals. The library's catalog is placed on microfiche, with copies available at many places, both inside the library and at other locations on campus. Other campus facilities include a modern student center and a recently-expanded bookstore. The Student Athletic Complex contains eight racquetball/handball courts, two squash courts, a twenty-five-meter swimming pool, and facilities for basketball, volleyball, badminton, tennis, weight training, and table tennis.

Georgia Tech conducts an extensive research program, with expenditures of $556 million annually. In addition, research conducted by teaching faculty, much work is done by the Georgia Tech Research Institute, non-profit applied research organization. One of the most successful research projects is the "SWEET Technology," which involves teaching courses in computer hardware, frequently interact with IC field faculty.

Several ICS faculty members are actively involved with Tech's Microelectronics Research Center, an organization created in 1981 to coordinate integrated circuit research being performed by Tech schools and the Georgia Tech Research Institute. A new $7.5 million building will soon be under construction to house the Research Center.

Another campus organization involved in high-technology research is the Advanced Technology Development Center, which not only attempts to attract high-technology firms to Georgia, but also provides low-cost space and advisory services to start-up companies and established firms opening new laboratories.

Atlanta

Atlanta, founded in 1837, is the largest city in the state of Georgia and the heart of the Atlanta metropolitan area, which includes nine counties and covers over 1,500 square miles. The city is home to over 3.5 million people, making it the 15th largest city in the country.

The city's location makes it a crossroads for all forms of transportation: highways, railways, and major airport hubs. Atlanta's airport is one of the busiest in the world, with over 100 million passengers passing through each year.

Atlanta is also known for its vibrant cultural scene, with world-class museums, theaters, and parks. The Atlanta Symphony Orchestra and the Alliance Theater are just two of the many cultural institutions that make Atlanta a thriving cultural center.

The Atlanta Symphony, under the direction of Robert Shaw, is one of the country's best orchestras. The Atlanta Ballet has drawn rave reviews from critics across the country. For opera fans, there is the annual visit by New York's Metropolitan Opera. Overall, Atlanta ranks high among American cities in almost every respect; in fact, the most recent Places Rated Almanac rated Atlanta the twelfth best metropolitan area in the nation.
Research

The School of Information and Computer Science maintains a vigorous program of research, made possible through corporate support and government grants. Research ranges from theoretical studies that pave the way for tomorrow's computer systems to practical efforts that shape the systems of today.

Major research efforts in the School include the Clouds project, the Software Test and Evaluation Project (STEP), and the projects of the artificial intelligence group. The goal of the Clouds project is the construction of a reliable distributed operating system. The STEP project, funded by the U.S. Department of Defense, is developing methodologies and tools for software testing. Projects underway within the School's AI group include two large AI systems: MEDATOR, for resolution of disputes, and PERSONA, for negotiation of labor contracts.

Although these projects are just a sample of ongoing research in the School, they convey a sense of the breadth of IOS research. These and other research projects are presented in more detail on the following pages.

Computer System Architecture

In the past, computer systems have been designed for efficient numerical computation, without regard for the kinds of algorithms that would later execute on them. Professor Pin-Yee Chen's research efforts are devoted to improving the design of computer systems by taking this type of information into account. In one research effort, he is developing analysis algorithms that when given a program, will construct a tree whose nodes contain information about the execution ordering for that program. This information about the computational structure of programs to be executed on a proposed computer system can then be used during the design of the system.

Professor Chen is also interested in the design of interconnection networks for efficient processing, alignment, and access to data. He is studying a number of proposed data communication networks and examining their suitability in the presence of multiplexing. The emphasis of his study is on the reliability and dynamic reconfiguration aspects of these networks.

Another of Professor Chen's interests is the relation between software and hardware in high-speed architecture. He is studying the interaction between modes of computation parallel, multiprocessor, and distributed, for example, and such important design issues as fork, join, synchronization, and communication overhead.

Programming Languages and Environments

The increasing sophistication of distributed computing systems mandates the development of tools that allow programmers to make efficient use of such systems. Having developed one such tool, a programming language that supports the design and construction of distributed programs, Professor Richard Leblanc and his students are now building a monitor that allows programmers to examine interactively the behavior of distributed programs. Monitoring a distributed program presents significant challenges, since the "state" of the program involves information about an arbitrary number of processes running on a number of machines. This problem is far more complex than monitoring a typical program on a single machine, where all of the state information is in a single address space.

The current version of the monitor collects information about process interactions as a program runs. It then uses a bitmap display to provide a high-level replay of the program's execution, showing creation and deletion of processes and messages passing between them. A programmer can interact with the display, controlling its speed and stopping it to look at messages. An interface to a single-process debugger is also provided, allowing the actual messages collected during an execution to be used as input during a debugging session on one process.

Professor Leblanc is also actively involved with the Clouds project (see "Distributed Operating Systems"). His interests in that project include the development of language features and programming methodologies appropriate for the Clouds environment. He is currently supervising the implementation of the Aequus language, which allows a programmer to define objects and actions. The features of Aequus allow access to the capabilities provided by the Clouds kernel. Work is currently in progress to develop programming techniques that use the features of Aequus to program replicated objects that can survive system crashes and other faults while providing high availability even while such problems are being reached.

A new faculty member, Professor M. W. Ahn, is developing communication structures for distributed languages. He is particularly interested in structures that are suitable for system level algorithms, such as those implemented as part of a distributed operating system. These algorithms often require multihost communication, hence it is a logical candidate for inclusion in a distributed language.
Software Engineering

Professor Richard DeMillo is investigating the properties of mutation analysis, a software evaluation technique. A software evaluator is a device to which one submits software and a collection of evidence purporting to show that the software performs as intended. With mutation analysis, the evidence used in the evaluation is a set of test cases, and the evaluation that is reported is an indication of how well the software has been tested. Such an evaluation represents a level of confidence in the adequacy of the test cases. To be useful as evidence of a program's correctness, test data need only distinguish the program from a finite number of alternatives—the alternatives that correspond to the most likely errors to be introduced by the programmer.

In a series of automated mutation analysis systems, researchers have explored the concept of choosing the set of alternatives by making simple mutations. The underlying assumption for such a choice has been that these mutations correspond to the errors most likely to be made in producing the given program.

One measure of how good the "mutation score" metric might be is how many "complex" mutants it leaves unexplained. An experiment to investigate this effect for COBOL programs, using representative COBOL and FORTRAN programs in the 100-1000 line range, has led to the conclusion that the mutation score comes very close to representing a probability of correctness. Further research in program mutation will proceed along three basic lines: statistical foundations, graphics, and display, and human factors.

Professor DeMillo's work in mutation analysis led him to establish the Software Test and Evaluation Project (STEP). The goal of STEP is to develop new policies and methodologies for use by the U.S. Department of Defense in the test and evaluation of software. Virtually all major defense systems contain computers as critical components; the testing of software is a key consideration for ensuring the reliability of these systems. Through the first two years of the project, Professor DeMillo's group carried out an exhaustive evaluation of software testing practice; this effort culminated in a number of recommendations for modifying policy for software testing. The next phase of STEP is to develop methodologies and tools that allow implementation of these policies. STEP is partially funded by DARPA, the new DoD Software Technology Initiative.

Distributed Operating Systems

Clouds is a global operating system that allocates and manages all resources, processors, files, etc., globally. No distinction is made between local and remote resources. Instead, decisions assigning processors to particular tasks are made on the basis of heuristics. Distributed data management is a major consideration for Clouds. The kernel provides highly reliable data management, simplifying this task for the operating system and application programs. The Clouds group is currently working on a prototype kernel for a cluster of UNIX 11/7000 systems.

The architecture supported by Clouds will provide objects and actions. These objects, which are instances of abstract data types, provide the basis for consistency. Atomic actions are the basis for recovery. Actions help to categorize the state of system components after failures.

The Cloud project is closely integrated with Professor LeBlanc's projects. His work on the monitoring of distributed programs will facilitate the development of systems under Clouds. Clouds, in turn, will provide support for distributed programming languages.

In another effort, Professor Mustapha Ahmed is developing an operating system-level multicast facility that will provide system calls that user processes can invoke to send and receive multicast messages. He is currently working on a multicast protocol and its implementation in UNIX. This protocol must deal with two problems. First, it must support message delivery to a group of processes that may execute on hosts connected to many networks. Second, the membership of a multicast group can change dynamically and hence a distributed algorithm is needed to maintain the membership of a multicast group. This is only one example. Professor Ahnem's interest in the correctness and performance of distributed algorithms, fault-tolerant algorithms, and specialized algorithms for various classes of applications. He has also observed that a general algorithm to implement a reliable multicast communication facility will be very expensive and complex. However, if the communication pattern of an application that makes use of a reliable multicast facility is known, a much more efficient algorithm is possible.
Computer Networks

Computer networking is fast becoming a major area of research within the school. Five faculty members perform research in the area, with the inauguration of the academic program in Computer Networking and Communication Systems. New courses will be offered in the fall semester.

During the past year, Professor Philip E. Rous and his colleagues have conducted an investigation of the performance of an Ethernet-type local area network. The objective of this project is to evaluate a CSMA/CD simulation model by measuring performance on an operational network. One of the most important findings was that currently available simulation equipment is not adequate at this time to properly model the performance of commercial communication systems or computer networks. Most of the existing instrumentation and test equipment is inadequate to perform this type of evaluation. The availability of this equip-
Computer Graphics

Fractals are an exciting new mathematical model. Professor Naylor's group has applied geometric modeling to create new ways of generating fractal objects, iterated function systems. Objects are created by defining them in terms of themselves. A set of affine transformations specifies this relation. A basic effort is the design of graphics systems that support concurrent processing. Work has focused on the organization of the refresh-frame buffer. Professor Naylor has devised a new scheme for distributing the pixels among the memory chips to allow easy concurrent access to any \( x \times M \) and \( M \times x \) block, where \( M \) is the total number of memory chips.

Another graphics project is a comparative experimental analysis of visible surface algorithms. Many such algorithms have been designed, but little is known about their relative performance. Professor Naylor's group has implemented ten visible surface algorithms and is now constructing initial experiments. The primary goal of this work is the identification of algorithms to be used in the design of high-speed VLSI-based architectures.

Other projects include the design of a generic modeling system based on the explicit use of functional programming at both the language level and in the actual representation of the geometric model. This allows every aspect of the model, including geometric shape and physical attributes, to be defined by functions. Thus, for instance, making aspects of a model a function of time, as is needed for animation, is straightforward.

Computer graphics research includes an Adage 2000 frame-buffer-based graphics system, an Evans & Sutherland PS 300 real-time vector system, a Tektronix 4155 system, and a Matrix 2000 camera system. The School's VAX 11/780 serves as the principal host for these systems.

VLSI Systems

The study of Very Large-Scale Integration systems is one of the most active areas of research in the School. Subjects of interest to the ICSI researchers are high-speed VLSI design, the application of VLSI components in computer system design, ICSI faculty currently involved in VLSI research are Professors DeMichio and Gopalakrishnan Vijayan.

One current thrust of the School's VLSI research is the development of design tools for high-speed chip design. Professor Vijayan was instrumental in the development of a Pascal-like language for procedural specification of layouts. An important aspect of the language is that the user is not required to specify any metric values, but instead describes only the topology of the circuit under design. Several chips have been successively designed and fabricated using this language. ICSI researchers have extended this approach to include a graphics interface for geometric programming, requiring the user to specify a little more than topology and cell organization. Tools such as these should lead to a significant reduction in the time required to design a VLSI layout.

Circuit reliability is an issue of overwhelming concern at all stages of the design process. ICSI research in this area involves two approaches. The first applies the techniques of mutation analysis to VLSI design testing. When a VLSI system is specified functionally, it is possible in principle to devise an automated system that will quickly and accurately assess the coverage of a given set of test cases for a circuit. Such techniques can be effective replacements for exhaustive testing, which is frequently not feasible. The fact that the techniques can be automated makes them suitable for inclusion in many stages of the design process.

Another important aspect of the ICSI effort is the involvement of ICSI faculty in the Institute-wide Microelectronics Research Center. Professors DeMichio and Bruce Naylor are members of its Technical Committee, while Professor Raymond Miller is on the Advisory Committee. The purpose of the Center is to coordinate the research and development of various circuits being performed by various units on the Tech campus, work that ranges from device physics and materials to system design and implementation. The Center sponsors seminars that allow microelectronic specialists on and off campus to be kept abreast of each other's research, and in addition, operates a technical information service for rapid dissemination of research results.

Although the Center is primarily a coordinating body, it runs a Small Micropower Design Lab to enhance the technical development of circuits. The Center also operates a Harris 8000/32-bit computer system.
Computer Security
Throughout most of the 1970s, secure computer systems were identified with secure processor multiplexing systems (e.g., Hydra, Multicore Unix). The conventional approach to the multilevel security problem for computer systems has been the software solution, exemplified by kernelized operating systems, which require the user to view security transactions through a small and tightly constrained operating system (the security kernel).
Professor Richard DeMillo, together with Professor Richard Lipton of Princeton University, is investigating the result of equaling the assumption that sharing implies multiplexing. His approach is motivated by the rapid advance of personal computers, distributed computing, and telecommunications technology.
When a single processor environment is replaced by a distributed system, designers can exploit the distributed nature of the system to enhance its security characteristics. In single-processor systems, a covert channel is a fuzzy, conceptual path for information flow. In a distributed system, however, it is possible to place a physical barrier between two processes merely by placing them on different processors. In the distributed system, the channel is explicit; it is a physical device that can be secured by encryption, monitored by physical means, and used to maintain various security properties by protocols.
The design of encryption-based protocols has led to theoretical questions concerning their analysis. Research is underway to provide models of cryptographic protocols in which security properties can be established. This work has already led to a number of new protocols, many of which are described in the book Applied Cryptology, Cryptographic Protocols, and Computer Security Models, written by Professor DeMillo in collaboration with four other experts in cryptography and security.

Theoretical Computer Science
Broadly speaking, theoretical computer science is concerned with the formulation of abstract models of computational processes and the study of properties of computation through these models. Researchers at Georgia Tech are currently studying not only standard sequential and parallel models of computation, but also newer models of VLSI computation, distributed computation, and database management.
One topic under study by Professor Richard DeMillo is the question of whether P = NP, which asks whether problems that can be solved efficiently on a nondeterministic model of computation can also be solved efficiently on a deterministic model.

He has been able to show that this question is formally equivalent to independence problems in classical mathematics. In addition to his work in "pure" theoretical computer science, Professor Miller is interested in using these models in other application areas as well.
A specialist in concurrent algorithms, Professor Gary Peterson has, in the past, examined such problems as mutual exclusion and resource control. More recently, he has been working on the design of efficient networks for parallel computation and efficient algorithms on networks, as well as studying the comparative power of concurrency synchronization mechanisms. He is also interested in other areas of theoretical computer science, including complexity theory and its relationship to information theory, models of VLSI computation, and extended models of alternation.

Professor Vepuri's interest in integrated circuits (see "VLSI Systems") affects much of his work in graph theory and algorithms. One of his research efforts is based on the fact that fast algorithms for obtaining near-optimal embeddings of certain classes of graphs can be used in the automation of the VLSI layout process. With Dr. William Knowlton at IBM San Jose, Professor Vepuri was able to characterize one such class of graphs—the rectilinear graphs—and obtain fast algorithms for recognizing and embedding these graphs. He is currently investigating efficient embeddings of planar graphs and other useful graph classes using various models for embeddings. He is also attempting to establish relations between planar geometry and planar graph theory.
Human Factors in Computer Systems

The objective of Professor Albert Badre’s research in the human factors of computer systems is to identify and explore empirically the human factors that affect the design and operation of computer systems and the efficiency of the software development process.

Professor Badre’s latest work involves designing transitions for the user interface. Many software systems that claim to be “user-friendly” merely provide an interactive, menu-driven interface. While this is fine for novices, more experienced users find menus to be an obstacle to efficient use of the system. Some systems attempt to solve this problem by providing both a novice, menu-driven interface and an expert, command-driven interface. Unfortunately, the two modes of interaction are usually independent; experience gained through the use of menus does not help the user learn the commands needed for the expert interface. The two-level interface also fails on two other counts: it does not allow a smooth transition from novice to expert, and it does not acknowledge that a person can simultaneously be an expert user of some system functions but a novice user of others.

Professor Badre believes that a user interface should be designed so that a novice user can gradually develop into an expert through experience gained by using the system. This goal can be achieved by designing the interface so that the following conditions are met:

1. Each task can be performed in two different modes: the novice mode and the expert mode. The novice mode should depend on recognition memory strategies, such as menus, prompts, and instructional screens, while the expert mode should depend on free recall of a language of interaction.

2. It should be possible to learn the expert’s vocabulary and syntax as a byproduct of practice as a novice. Linguistic continuity should be maintained across modes.

3. The expert’s language of interaction should allow the bypassing of novice interactive strategies. It should also allow the introduction of new vocabulary and the generation of new sentences for “chunking” actions grouping atomic actions into larger units.

In another project, Professor Badre is studying the organizing, sequencing, and updating of information on computer displays. His research group has conducted experiments on the identification and characterization of the user’s functional information needs, with particular attention to the effect on the user’s chunking or organizing of information display, the availability of variables and information presentation and display. The goal is to develop an algorithm for interacting with a computer system in order to compare it with an alternative on-screen interface. His prototype system is under development.

Organizing and remembering events as people might do.

- In current work, she addresses the problems of how such a memory can be used in problem solving, how task competence changes through experiences, and how experience enhances problem-solving capabilities. The ultimate goal of this research is to produce computer systems capable of not just expert reasoning, but reasoning in such a way that the program can adapt and improve over time.

In studying the role of experience in clinical problem solving, Professor Kolodner has examined how experience changes the knowledge used in diagnosing and how experience is used during diagnosis. Failure to diagnose correctly, for example, can be avoided if a clinician is reminded of a previous similar case. The diagnosis made in that case serves as a hypothesis in the current case. The long processes that might have been necessary to track down a correct diagnosis will be shorter considerably the second time. The group has designed an algorithm for diagnosis that includes experience.

As a first step, Professor Kolodner designed CYRUS, a computer system that stores events from the lives of former Secretary of State Eugene V. March and Edmond Muskie and asks questions when viewing in natural language. CYRUS is based on a human model, organizing and remembering events as people might do.

- In current work, she addresses the problems of how such a memory can be used in problem solving, how task competence changes through experiences, and how experience enhances problem-solving capabilities. The ultimate goal of this research is to produce computer systems capable of not just expert reasoning, but reasoning in such a way that the program can adapt and improve over time.

In studying the role of experience in clinical problem solving, Professor Kolodner has examined how experience changes the knowledge used in diagnosing and how experience is used during diagnosis. Failure to diagnose correctly, for example, can be avoided if a clinician is reminded of a previous similar case. The diagnosis made in that case serves as a hypothesis in the current case. The long processes that might have been necessary to track down a correct diagnosis will be shorter considerably the second time. The group has designed an algorithm for diagnosis that includes experience.

As a first step, Professor Kolodner designed CYRUS, a computer system that stores events from the lives of former Secretary of State Eugene V. March and Edmond Muskie and asks questions when viewing in natural language. CYRUS is based on a human model, organizing and remembering events as people might do.
Professor Richard Cullingford's long-term research aim is the design of intelligent conversational systems capable of robust natural language interaction with users in a variety of application domains. For the last decade, he has been evolving methodologies, software tools, and architectures for language analysis and generation, common-sense reasoning and inference, knowledge base management, and conversational control. His experience is summarized in the book Natural Language Processing: A Knowledge Engineering Approach, to be published by Rowman & Allanheld in late 1985. The primary testbed for his research has been the ACE (Academic Counseling Experiment) system, a conversational program that simulates a faculty advisor interacting with students in tasks such as question answering concerning course offerings, preregistration and establishing a plan of study, and explaining the curriculum. The focus of the ACE research has been on the control of the language processing, reasoning, and memory retrieval resources of an intelligent system to manage the tremendous complexity that appears in realistic conversational behavior. The directing idea has been to give an intelligent system its own internal goals, so that in some reasonable sense it has a need to interact with its users. A current aim is to bring up a version of ACE that can interact with real students.

The main problems to be solved in the realization of robust language-using systems are not in language processing, per se, but in the design of the system's knowledge base and the applicability of that knowledge in problem solving. A current focus of attention is in common-sense reasoning, an attempt to model how people use their knowledge of the world, including especially contingent reasoning, in their day-to-day activities. An appealing model of this process is to view reasoning as a species of search, in which portions of a virtual belief net are utilized as the reasoner runs. Professor Cullingford is currently implementing a Best First Reasoner, which attempts to apply both general-purpose and domain-specific meta-rules to select the currently most plausible reasoning chain in mind. The Reasoner is a resource of a simulated planar non-time monitor for a robot system attempting to navigate in a town using a faulty street map. The ICSI artificial intelligence group currently includes ten graduate students. Programming is done in LISP on Symbolics LISP machines.

Cognitive Science

Cognitive science is a newly emerging discipline that combines techniques and approaches from the fields of artificial intelligence, linguistics, psychology, anthropology, and philosophy to pursue the problem of cognition knowledge use and acquisition. ICSI researchers approach cognitive science from the perspectives of artificial intelligence, human information processing, and language processing. Professor Janet Kolodner is involved in cognitive science research as a part of her work in artificial intelligence. Like that work, her cognitive science efforts explore the role of experience in expert and common-sense reasoning. In addition to the work described in the previous section, she and Professor Larry Barsalou, a psychologist at Emory University, are testing some of the predictions arising from the CYRUS model of long-term memory. Their work involves both psychological experiments and experiments using the computer model CYRUS. Along with a psychology postdoctoral fellow, she is preparing to do in-depth interviews with novice and expert problem solvers to learn more detail what different experience plays in human problem solving. A recently initiated project, an advisory system for the acquisition of household products, will serve as another vehicle for investigating memory and problem-solving phenomena.

Professor Albert Bacce has conducted research in the area of problem-solving strategies, with the goal of achieving a better understanding of the processes underlying the skilled solving of ill-defined problems. Those processes include both problem-solving strategies and strategies for acquiring and organizing the knowledge necessary for solving problems. The effective definition of such processes would lead, on the one hand, to the development of computer programs that model expert solving of ill-defined problems, and on the other hand, to the computer extension of human information-handling and problem-solving capabilities. Professor Bacce has been successful in introducing new techniques to delineate the representational structures in problem-solving protocols.
Systems Theory
Professor Pranas Zunde is conducting a study of system structures and decomposition, in which global characteristics of various properties of linear dynamical systems, including linear automata, such as controllability, reachability, observability, and reconstructability, are developed using an algebraic approach (Shannon). He has been able to relate properties of state space lattices to characteristics of these systems' parameter matrices and develop a method for selection of a system model with desirable structural properties. He is extending his results to state feedback control systems, characteristic of state space structure of bi-linear systems, and system decomposition problems.

Information Science
Professor Pranas Zunde's work in information science focuses on the empirical foundations of the subject. He is attempting to identify the body of knowledge that can be derived from empirical observations of information phenomena and is representative of information science as a distinct scientific discipline. The laws and hypotheses under study include information value, relevance, aging of information, growth of knowledge, obstacles, information scatter, productivity of information sources, and information flow.

In one project, Professor Zunde is investigating the feasibility of operational definitions of Zipf's "least effort" principle and its application as an optimality criterion in information systems design. In another project, he is examining certain quantitative aspects of semantic coding in an attempt to determine the relationship between diversity of meaning and syntactic properties of words and morphemes in informative texts. He will test entropy maximization theory as well as other relevant theories for their capability to account for, explain, and predict empirical findings of the study.

Finally, Professor Zunde will determine the feasibility of developing a comprehensive theory of semantic coding along the lines of Shannon's information theory, using the empirical results of the former investigation as test data. The study may have applications in such areas as information retrieval and the modeling and design of interactive computer systems.

Professor Zunde is also working in the field of pattern recognition, where he is currently studying adaptive methods of feature extraction. Present methods for automatic extraction and selection of features from data for pattern recognition and classification are operationally limited to data types that seldom are of interest in real-life applications. One of the reasons for the limited use of present methods is their insensitivity to changes in the characteristics of patterns (e.g., input data). To overcome this limitation, Professor Zunde is examining adaptive systems of feature analysis and extraction, with emphasis on binary patterns and application of error-correcting coding techniques.

Information Systems
Professor Vladimir Slamecka directs a four-year $3.3 million effort to design and implement a national network of scientific and technical information services for the Arab Republic of Egypt. The major product of the firm's completed phase of the program was the structured design of this totally oriented information network. The next phase includes the development and implementation of these services involving, among others, the development of a distributed network of Egyptian databases, interactive searching of foreign databases, a document delivery system, a nationwide manpower training program, and the establishment of the legal and organizational superstructure to coordinate and govern this evolving national network.

Professor Slamecka's program addresses a number of R&D issues, including national portability of software environments, bi-lingual end-user software, uniform database design for full-text and faceted data, and a standard bilingual command language interface for databases of diverse content. The program is receiving international attention as a case study demonstrating innovative application of state-of-the-art information technology in developing countries. Professor Slamecka's work on the program was cited by the American Association for the Advancement of Science when it elected him a Fellow in 1983.
Computer-Supported Instruction

The School's research program in computer-supported instruction received a boost in 1984 with the opening of the IBM Computer-Supported Instruction Laboratory. The goals of the Laboratory, which is one of four national demonstration sites for IBM's Academic Information Systems Group, are to demonstrate the use of IBM systems for instruction, evaluate them, and develop courseware. The Laboratory includes two classrooms, each with thirty IBM PC student workstations and one instructor station, and a demonstration room containing a variety of personal computers. All of the PCs are networked to an IBM 4381 via two IBM Series 1 systems. A third Series 1 gives an instructor access to the 4381.

The CSI Laboratory is operated by the School of IC&S and used by both IC&S and other departments, including Chemistry, City Planning, English, Mathematics, Modern Languages, Physics, and Social Sciences for purposes that range from classroom support of technical writing to preparation of sophisticated artificial intelligence projects. Lucia Chirivigi, associate director of the School, and Luis Karlqvist, dean of the College of Sciences and Liberal Studies, are co-directors of the Laboratory. During the 1984-85 academic year, thirty-nine courses in ten different disciplines used the Laboratory with total student enrollment exceeding 3500.
Technical Reports
July 1994 to June 1995

Gifts, Grants, and Contracts

Programs

Undergraduate

Graduate

The School of Information and Computer Science offers programs of study leading to the M.S. and Ph.D. degree in Information and Computer Science. Students holding a bachelor's degree in computer science or other quantitative field may be admitted to the M.S. degree program. The M.S. program accommodates students with a variety of career objectives, including technical and management positions in industry and government.

The doctoral program prepares exceptionally qualified students for research, academic, and policy/market positions. Students applying for admission to the doctoral program must offer evidence of exceptional scholastic ability, intellectual creativity, and research motivation.

Further information and application forms may be obtained by writing to:
Mr. Edmond F. Furmano, Assistant to the Director
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, Georgia 30332-0030


National Science Foundation, M. S. McInerney, Academic Applications Architecture for Reliable Distributed Computing, 1994, 91-109R, $45.00.

National Science Foundation, R. E. Miller, A. Balle, R. J. Lefebvre, and M. S. McInerney, Education for Reliability, Integrated Distributed Computing, 1994-1995, $100.00.


Undergraduate Courses

IC 4136: Problem Solving
IC 4155: Introduction to Theory of Computing
IC 4202: Computer Organization and Programming
IC 4203: Computer Organization and Programming II
IC 4204: Introduction to Database Systems
IC 4205: Introduction to Mathematical Logic
IC 4206: Introduction to Theory of Computing
IC 4207: Introduction to Software Development
IC 4208: Introduction to Artificial Intelligence
IC 4209: Artificial Data Processing
IC 4210: Survey of Programming Languages
IC 4211: Computer- Oriented Numerical Methods
IC 4212: Introduction to Mathematics, Logic, and Numerical Methods
IC 4213: Introduction to Information Processing

Graduate Courses

IC 4136: Problem Solving
IC 4155: Introduction to Theory of Computing
IC 4202: Computer Organization and Programming
IC 4203: Computer Organization and Programming II
IC 4204: Introduction to Database Systems
IC 4205: Introduction to Mathematical Logic
IC 4206: Introduction to Theory of Computing
IC 4207: Introduction to Software Development
IC 4208: Introduction to Artificial Intelligence
IC 4209: Artificial Data Processing
IC 4210: Survey of Programming Languages
IC 4211: Computer- Oriented Numerical Methods
IC 4212: Introduction to Mathematics, Logic, and Numerical Methods
IC 4213: Introduction to Information Processing

Current Doctoral Students

Sub Y Cho
Database systems, artificial intelligence, software engineering
Michael G. Christel
Computer graphics, artificial intelligence
Stephen D. Crawford
Computer graphics, artificial intelligence
David A. Crick
Programming languages, natural language processing, algorithms
Ramesh R. Komuravelli
Artificial intelligence, cognitive science
Gema V. Iliev
Artificial intelligence, system design
Martin E. Davies
Software engineering, computer security
J. Dana Eckert
Software engineering, decision-making
Tom C. Eshleman
Artificial intelligence, theoretical computer science
Heinz W. Dierks
Artificial intelligence, computer science
Anchalee P. Fruin
Artificial intelligence, computer science
Marina S. Garcia
Intelligent computer-aided instruction, expert systems
Gail R. Gilbreath
Artificial intelligence, artificial neural networks
Michael I. Gasser
Artificial intelligence, artificial neural networks
R. C. Gardner
Artificial intelligence, computer science
Jane A. Gerber
Artificial intelligence, artificial neural networks
N. Glenn Glover
Operating systems, computer architecture, artificial intelligence
Robert A. Hernandez
Operating systems, computer architecture
Elise M. Hill
Artificial intelligence
Laurel B. Hodges
Computer graphics, human factors
Nancy K. Carson
Computer graphics, human factors
Rose L. Haas
Artificial intelligence
Yung K. Hsu
Expert systems
Arto T. Korpilahti
Data communications
R. D. Keller
Operating systems, database systems
Carol E. Kilpatrick
Database systems, artificial intelligence
David J. Pitts
Database systems, artificial intelligence
Ramesh R. Komuravelli
Artificial intelligence, cognitive science
Scott A. Whittaker
Artificial intelligence, artificial neural networks
C. Thomas Wilkes
Programming methodologies for reliability and availability
Kevin L. Smith
Large scale systems
David C. Sowell
Programming languages and environments, operating systems
Eugene K. Sparrow
Programming languages and environments, operating systems
N. Glenn Glover
Operating systems, computer architecture
Laurel B. Hodges
Computer graphics, human factors
R. C. Gardner
Artificial intelligence
Jane A. Gerber
Artificial intelligence
Nancy K. Carson
Computer graphics, human factors
Rose L. Haas
Artificial intelligence
Yung K. Hsu
Expert systems
Arto T. Korpilahti
Data communications
R. D. Keller
Operating systems, database systems
Carol E. Kilpatrick
Database systems, artificial intelligence
David J. Pitts
Database systems, artificial intelligence
Ramesh R. Komuravelli
Artificial intelligence, cognitive science
Scott A. Whittaker
Artificial intelligence, artificial neural networks
C. Thomas Wilkes
Programming methodologies for reliability and availability
Kevin L. Smith
Large scale systems
David C. Sowell
Programming languages and environments, operating systems
Eugene K. Sparrow
Programming languages and environments, operating systems
N. Glenn Glover
Operating systems, computer architecture
Laurel B. Hodges
Computer graphics, human factors
R. C. Gardner
Artificial intelligence
Jane A. Gerber
Artificial intelligence
Nancy K. Carson
Computer graphics, human factors
Rose L. Haas
Artificial intelligence
Yung K. Hsu
Expert systems
Arto T. Korpilahti
Data communications
R. D. Keller
Operating systems, database systems
Carol E. Kilpatrick
Database systems, artificial intelligence
David J. Pitts
Database systems, artificial intelligence
Ramesh R. Komuravelli
Artificial intelligence, cognitive science
Scott A. Whittaker
Artificial intelligence, artificial neural networks
C. Thomas Wilkes
Programming methodologies for reliability and availability
Kevin L. Smith
Large scale systems
David C. Sowell
Programming languages and environments, operating systems
Eugene K. Sparrow
Programming languages and environments, operating systems
N. Glenn Glover
Operating systems, computer architecture
Laurel B. Hodges
Computer graphics, human factors
R. C. Gardner
Artificial intelligence
Jane A. Gerber
Artificial intelligence
Nancy K. Carson
Computer graphics, human factors
Rose L. Haas
Artificial intelligence
Yung K. Hsu
Expert systems
Arto T. Korpilahti
Data communications
R. D. Keller
Operating systems, database systems
Financial Aid for Graduate Students

The forms of financial aid available to graduate students in the School of Information and Computer Science are listed below. Other kinds of financial assistance are offered by Georgia Tech, e.g., Veterans Administration Program, Work-Study, State Loans, National Direct Student Loans, and short-term loans. Information on these is available from:
Office of Financial Aid
Georgia Institute of Technology
Atlanta, Georgia 30332-0460

President's Fellowships
Stipend of $16,000 for twelve months plus waiver of tuition and fees for one year. The award can be extended for two additional years, based on the student's evaluation of the student's academic performance and research potential. Second and third-year awards provide a stipend of $5,000, a waiver of tuition and fees, and a supplemental research or teaching assistantship to provide the student with a total income comparable to the first-year award. Awarded to first-year Ph.D. and exceptional M.S. students by the Dean of Graduate Studies upon recommendation of the School. Highly competitive.

Graduate Teaching Assistantships
Part-time employment (usually one-third or half-time) at a corresponding fraction of the full-time rates ($20,000 and $25,000 for twelve months) plus waiver of all tuition and fees. Selection based on the applicant's ability to contribute to research projects of the School. Normally open only to Ph.D. students.

Graduate Research Assistantships
Part-time employment (usually one-third or half-time) at a corresponding fraction of the full-time rates ($20,000 and $25,000 for twelve months) plus waiver of all tuition and fees. Prospective graduate students should apply for competitive national fellowships such as those offered by the National Science Foundation.

Regents' Opportunity Scholarships
Stipend of $5,000 for twelve months. Awards are made to economically disadvantaged residents of Georgia.

Out-of-State Tuition Waivers
Awarded by the Dean of Graduate Studies upon recommendation by the School. Selection is based on academic performance.

Federal Fellowships and Traineeships
Subject to the availability of funds, a limited number of fellowships and traineeships may be awarded through participation in programs sponsored by agencies of the federal government.

For More Information
The following publications provide more information on the educational programs of the School of Information and Computer Science:
- Doctoral Program in Information and Computer Science
- Programs and Courses, School of Information and Computer Science

General Catalog, Georgia Institute of Technology
All are available from:
Mr. Edmond F. Rizzano, Assistant to the Director
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, Georgia 30332-0280
For more information about the Special Program in Computer Networking and Communications Systems, write to:
Professor Philip H. Franks, Jr.
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, Georgia 30332-0280
The School publishes many technical reports each year. For an up-to-date list of available reports, write to:
Ms. Karen Hutchison
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, Georgia 30332-0280