COMPUTER SCIENCE

www.cs.siu.edu
csinfo@cs.siu.edu

COLLEGE OF SCIENCE

Carver, **Norman F**., **III**, Associate Professor, Ph.D., University of Massachusetts, 1990; 1995. Multi-agent systems, sensor interpretation, knowledge-intensive control of AI systems.

Che, Dunren, Assistant Professor, Ph.D., Beijing University of Aeronautics and Astronautics, Beijing China, 1994; 2001. Database, structured document management, bioinformatics.

Danhof, **K**. **J**., Professor, *Emeritus*, Ph.D., Purdue University, 1969; 1969.

Gupta, **Bidyut**, Professor, Ph.D., University of Calcutta, 1986; 1988. Distributed systems, fault-tolerant computing, mobile communication.

Hou, **Wen-Chi**, Associate Professor, Ph.D., Case Western Reserve University, 1989; 1989. Statistical databases, query optimization, data stream processing, spatial data structures.

Mark, **Abraham M**., Professor, *Emeritus,* Ph.D., Cornell University, 1947; 1950.

McGlinn, **Robert J**., Associate Professor, Ph.D., Southern Illinois University Carbondale, 1976; 1981. Computer education, social issues of computing, software engineering, object-oriented programming.

Mogharreban, Namdar, Assistant Professor, Ph.D., Southern Illinois University Carbondale, 1989; 1999. End user computing, computer training, computing in special populations, decision making, decision science.

Phillips, **Nicholas C**.**K**., Associate Professor, *Emeritus,* Ph.D., University of Natal, 1967; 1988.

Rahimi, Shahram, Assistant Professor, Ph.D., University of Southern Mississippi, 2002; 2002. Distributed computing, software agents, expert systems, fuzzy logic, soft computing.

Wainer, **Michael S**., Associate Professor, Ph.D., University of Alabama at Birmingham, 1987; 1988. Computer graphics, agile software development and testing, HCI.

Wang, Chih-Fang, Assistant Professor, Ph.D., University of Florida, 1998; 1999. Design and analysis of algorithms, data structures, parallel algorithms, high-performing computing, optical interconnection networks.

Wright, **W**. **E**., Professor, *Emeritus*, D.Sc., Washington University, 1972; 1970.

Zargham, **M**. **R**., Professor and *Interim Chair*, Ph.D., Michigan State University, 1983; 1983. Computer architecture, fuzzy logic, neural networks, parallel processing, expert systems.

The Department of Computer Science offers a graduate program leading to the Master of Science degree with a major in computer science. Application forms for admission to this degree program must be obtained from the department. Please contact the department by postal mail or email (csinfo@cs.siu.edu) to request an application packet or make inquiries.

A non-refundable application fee of $35.00 must be submitted with the application. Attach your check or money order, payable to Southern Illinois University Carbondale, to the top of the application form. Do not send cash. Only checks or money orders in U.S. funds will be accepted.

Admission and Retention

Decisions concerning the admission of students to, and retention of students in, the graduate program will be made by the department faculty subject to the requirements of the Graduate School.

The evaluation of applicants for admission is based primarily on the student's academic record with particular attention being given to past performance in relevant undergraduate course work. Applicants are expected to have a substantial background in undergraduate computer science courses covering high level and assembly language programming, data structures, computer organization, logic design as well as discrete mathematics, calculus, and linear algebra. The applicant is expected to have completed course work in the above subject areas prior to admission. Normally, a GPA of at least 3.0/4.0 is required by the Department of Computer Science. In addition, applicants must submit Graduate Record Examination (GRE) general test scores. It is recommended that results from the GRE subject area test (computer science or a related field) also be submitted.

Requirements. A student who has been admitted to the graduate program in computer science can meet the requirements for the Master of Science degree by completing 33 hours of graduate credit subject to the following constraints:

1 Each of the courses CS 401, CS 414, and CS 455 must be taken. (If a specific course, or its equivalent, is already part of the student's academic background, an alternate course will be substituted, with Graduate Program Director approval.)
2. The 33 hours of graduate work must include at least four 500-level CS lecture courses.
3. Elective courses should be chosen from CS courses, excluding any cross-listed courses. (Alternate or substitution courses require Graduate Program Director approval.)
4. Students are required to write a thesis carrying 6 credit hours under CS 599. No more than 3 credit hours of CS 599 can be taken per semester.
5. All course registrations must have the written approval of the Graduate Program Director.

After completion of all work, the student will be given a final oral examination over the thesis and other course work.

Courses (CS)

401-3 Computer Architecture. Review of logical circuit design. Hardware description languages. Algorithms for high-speed addition, multiplication and division. Pipelined arithmetic. Implementation and control issues using PLA's and microprogramming control. Cache and main memory design. Input/Output. Introduction to interconnection networks and multiprocessor organization. Prerequisite: 315 with a grade of *C* or better.

402-3 Theory and Applications of Computer Aided Design. A study of algorithmic techniques, which solve high complexity design rules. Graph algorithms and formulations, randomized solutions, techniques from operations research and statistics, computational geometry algorithms and data structures are introduced. The techniques are mainly applied on the physical design/automation problem for integrated circuits and systems. Prerequisite: 315 and 355 each with grade of *C* or better.

414-3 Operating Systems. An extended treatment of the components of operating systems, including I/O programming, memory management, virtual memory, process management, concurrency, device management and file management. Prerequisite: 306 and 330 each with a grade of *C* or better.

416-3 Compiler Construction. Introduction to compiler construction. Design of a simple complete compiler, including lexical analysis, syntactical analysis, type checking and code generation. Prerequisite: 306 and 311 each with a grade of *C* or better.

420-3 Parallel and Distributed Computing. This course serves as an introduction to the areas of parallel and distributed computing. The major approaches to parallel programming, including shared-memory multiprocessing and message-passing multicomputing, will be covered in some detail. Students will have programming experience in each of these paradigms. Architectural considerations, algorithm design and measures of performance will be covered. In addition, the course will provide an introduction to distributed computing on a network of computers. Parallel and distributed computing will be contrasted. Other approaches to parallelism including data parallelism (SIMD) and vector processing will be surveyed. Prerequisite: 306 and 355 each with a grade *C* or better.

430-3 Database Systems. A comprehensive treatment of database systems, including network, hierarchical and relational systems. Prerequisite: 330 with a grade of *C* or better.

432-3 File Organization. Secondary storage device. File designs and algorithms for efficient storage, retrieval and updating of information in secondary memory. Space and time analysis. Prerequisite: 330 with a grade of *C* or better.

435-3 Software Design and Development. An exercise in the analysis, design, implementation, testing, and maintenance of a large modular application system. Team production of a system is the focal point for the course. Topics include the system life cycle, system specification, human interfaces, modular design, improved programming techniques and program verification and validation. Prerequisite: 306 and 330 each with a grade of *C* or better.

436-3 Artificial Intelligence I. Search and heuristics, problem reduction. Predicate calculus, automated theorem proving. Knowledge representation. Applications of artificial intelligence. Parallel processing in artificial intelligence. Prerequisite: 311 and 355 each with a grade of *C* or better.

440-3 Computer Networks. Design and analysis of computer communication networks. Topics to be covered include queuing systems, data transmission, data link protocols, topological design, routing, flow control, security and privacy and network performance evaluation. Prerequisite: 306, 315 and 330 each with a grade of *C* or better.

447-3 Introduction to Graph Theory. (Same as Mathematics 447.) Introduction to theory of graphs, digraphs, and networks and applications to electrical systems and computer science. Topics include blocks and cutpoints, Eulerian graphs, trees, cycle and cocycle spaces, planarity and Kuratowski's Theorem, connectivity and Menger's Theorem, Hamiltonian graphs, colorability and Heawood's Theorem, flows in networks and Ford-Fulkerson Theorem, critical path analysis. Prerequisite: Mathematics 349 or consent of instructor.

449-3 Introduction to Combinatorics. (Same as Mathematics 449.) An introduction to combinatorial mathematics with computing applications. Topics include selections and arrangements, generating functions, recursion, inclusion and exclusion, coding theory, block designs. Prerequisite: Mathematics 349 or consent of instructor.

451-3 Theory of Computing. The fundamental concepts of the theory of computation including finite state acceptors, formal grammars, Turing machines and recursive functions. The relationship between grammars and machines with emphasis on regular expressions and context-free languages. Prerequisite: 311 and 355 each with a grade of *C* or better or graduate standing.

455-3 Design and Analysis of Computer Algorithms. An extensive treatment of the design, analysis and complexity of algorithms. Lower bound arguments, divide-and-conquer techniques, greedy algorithms, dynamic programming, graph theoretic algorithms, PRAM algorithms, and NP-completeness and approximation algorithms. Prerequisite: 355 with a grade of *C* or better or graduate standing.

471-3 Introduction to Optimization Techniques. (Same as Mathematics 471.) Nature of optimization problems. General and special purpose methods of optimization, such as linear programming, classical optimization, separable programming, integer programming and dynamic programming. Prerequisite: 202 and Mathematics 221 and 250.

472-3 Linear Programming. (Same as Mathematics 472.) Nature and purpose of the linear programming model. Development of the simplex method. Application of the model to various problems. Duality theory. Transportation. Assignment problem. Postoptimality analysis. Prerequisite: 202 and Mathematics 221.

475-6 (3, 3) Numerical Analysis. (Same as Mathematics 475.) An introduction to the theory and practice of computation with digital computers. Topics include the solution of nonlinear equations, interpolation and approximation, solution of systems of linear equations, numerical integration, solution of ordinary differential equations, computation of eigenvalues and eigenvectors and solution of partial differential equations. Prerequisite: **(a)**

Mathematics 221 and 250 and Computer Science 202 or equivalent programming proficiency; **(b)** Mathematics 305 and Computer Science 464a.

484-3 User Interface Design and Development. Human-computer interaction and the importance of good interface design. Interface quality and methods of evaluation. Interface design examples and case studies. Prototyping and implementation techniques. Task analysis and the iterative design cycle. Dialogue techniques, basic computer graphics, I/O device, color and sound. Use of at least one interface toolkit and development methodology to complete an interface design project. Prerequisite: 306 with a grade *C* or better.

485-3 Computer Graphics. Study of the devices and techniques for the use of computers in generating graphical displays. Includes display devices, display processing, transformation systems, interactive graphics, 3-dimensional graphics, graphics system design and configuration, low and high level graphics languages and applications. Prerequisite: 306 with a grade of *C* or better; Mathematics 150 and 221 are recommended.

490-1 to 6 (1 to 3 per semester) Readings. Supervised readings in selected subjects. Prerequisite: consent of instructor and department.

491-1 to 4 Special Topics. Selected advanced topics from the various fields of computer science. Prerequisite: consent of instructor.

492-1 to 6 (1 to 3 per semester) Special Problems. Individual projects involving independent work. Prerequisite: consent of department.

493-1 to 4 Seminar. Supervised study. Preparation and presentation of reports. Prerequisite: consent of instructor.

501-3 Advanced Computer Architecture. Hardware and software elements of multiprocessors, multicomputers, pipeline and array machines, data flow architecture and other state-of-the-art architectures. Design principles related to machine structures, interconnection networks, control software and hardware, data storage and access. Prerequisite: 401.

502-3 Design and Analysis of VLSI Systems. This course covers the theory, technology, fabrication and design of digital integrated circuits as they are commonly used in modern digital computers. The topics covered include techniques for solving problems occurring in VLSI and ULSI layouts, built-in self-testing, design for testability and logic synthesis. The course also treats additional selected advanced topics. Prerequisite: 401 and either 402 or consent of instructor.

503-3 Fault-Tolerant Computing Systems. An introduction to different aspects of fault-tolerance in computing systems. Concurrent checking techniques. Redundancy techniques. Evaluation methods. System-level diagnosis and fault-tolerant VLSI architectures. Prerequisite: 401.

504-3 Testing of Integrated Circuits and Systems. This course provides a detailed treatment of digital systems testing and testable design. Topics covered include fault modeling, fault simulation, testing for stuck faults, testing for bridging faults, delay faults, IDDQ faults, functional testing, built-in testing, design for testability, logic and system level diagnosis and PLA testing. Prerequisite: 401 and either 402 or consent of instructor.

511-3 Formal Specification of Programming Languages. A survey of modeling techniques and Meta languages for the formal specification of the syntax and semantics of high-level programming languages. Prerequisite: 311.

512-3 Declarative Programming. An advanced level course on nonprocedural programming with emphasis on logic programming, pure functional programming, and the characteristics of the declarative style common to these two paradigms. Topics include logic programming, functional programming, implementation consideration for each along with current research topics in the areas. Prerequisite: 311.

514-3 Advanced Operating Systems. Rigorous treatment of advanced topics in operating systems. Multiprocessor and distributed operating systems. Highly concurrent machines. Performance analysis of memory management and scheduling algorithms. Security in operating systems. Prerequisite: 414.

516-3 Advanced Compilers. A continuation of 416 including advanced topics in lexical and syntax analysis, error recovery, sematic analysis, code optimization and compiler compilers. Prerequisite: 416.

520-3 Advanced Topics in Parallel & Distributed Computing. An advanced treatment of parallel and distributed computing; review of hardware and software considerations for parallel computation; development and analysis of parallel algorithms (with particular attention to the communication and synchronization costs associated with parallel algorithms); effect of granularity on performance; a comparison of the parallel and distributed programming paradigms including a detailed study of the central features of each approach; software systems for distributed computing including exposure to one or more distributed programming environments; the direction of parallel computing as suggested by recent, high level parallel languages; parallelizing serial programs; parallelizing compilers; future directions of parallel and distributed computing systems. The course will include a student project. Prerequisite: 420.

530-3 Advanced Data Base System. A detailed treatment of advanced topics in data base systems including, but not limited or restricted to, relational database theory, query optimization, recovery techniques, concurrency control, distributed database systems, security and integrity and database machines. Prerequisite: 430.

532-3 to 6 Topics in Information Systems. A detailed study of two or three topics relevant to information systems. Topics may include but are not limited to sorting, searching, information retrieval and automatic text processing, database security and encryption, distributed databases and data communication. Prerequisite: 430 and consent of instructor.

536-3 Artificial Intelligence II. Theorem proving, the Resolution Principle, strategies, and achievements. Program verification. Natural language processing. Other selected topics. Prerequisite: 436.

553-3 Formal Languages and Automata. The Chomsky hierarchy of formal grammars and the corresponding classes of automata. Turing machines and basic concepts of computability. Recursive and recursively enumerable languages. Closure properties. Undecidable problems about Turing machines and context-free languages. Deterministic context-free languages and the construction of LR parsers. Prerequisite: 451.
555-3 Computability and Complexity. Turing machines and other models of computation. Computable functions. Church's thesis. Solvable and unsolvable problems. Introduction to complexity theory including the classes P and NP. Polynomial time approximation algorithms for NP-complete problems. Prerequisite: 451.
564-1 to 12 Advanced Topics in Numerical Analysis. (Same as Mathematics 572.) Selected advanced topics in Numerical Analysis chosen from such areas as: approximation theory; numerical solution of initial value problems; numerical solution of boundary value problems; numerical linear algebra; numerical methods of optimization; functional analytic methods. Prerequisite: consent of instructor.
570-3 to 9 per topic (3,3,3) Topics in Operations Research. (Same as Mathematics 570.) **(a)** Netflows. Builds on network and generalized network models for the transportation, transshipment, assignment, shortest path, and maximal flow. Prerequisite: 472 or Mathematics 472. **(b)** Advanced computer simulation. Review of GPSS. Advanced topics in GPSS. Generation of random variates. Validation, parametric, and nonparametric tests. Design of experiments, optimization, parameter tuning. Analysis of variance, spectral analysis, and variance reduction. Prerequisite: 470 and Mathematics 480 or 483. **(c)** Large scale linear programming. Advanced L.P. techniques for sparge matrices and reinversion routines. Prerequisite: 472 or Mathematics 472. **(d)** Nonlinear programming. Integer programming with branch and bound and cutting plane methods for solving integer-programming problems. Basic dynamic programming with emphasis on the methods and applications. Prerequisite: 472 or Mathematics 472.
585-3 Advanced Topics in Computer Graphics. Study of computer graphics for realistic image synthesis. Object modeling and associated date structures. Advanced rendering techniques such as raytracing and radiosity. Efficiency considerations. Image composition and compression. Current advances and research problems in realistic computer graphics. Prerequisite: 485.
586-3 Pattern Recognition and Image Processing. An introduction to the area of computer vision for the purpose of restoration, segmentation, encoding, analysis and recognition of pictures. Topics include: image transforms, edge detection, smoothing, filtering, pseudo-coloring, syntactic methods in scene analysis, parametric decision theory, non-parametric decision theory, linear discriminant functions, parameter estimation, supervised learning and unsupervised learning. Prerequisite: 220 and Mathematics 380 or consent of instructor.
590-1 to 9 Readings. Supervised readings in selected subjects. Graded *S/U* only. Prerequisite: consent of instructor and department.
591-1 to 9 (1 to 3 per topic) Special Topics. Selected advanced topics from the various fields of computer science.
593-1 to 4 Seminar. Preparation and presentation of reports. Graded *S/U* only. Prerequisite: consent of instructor.
599-1 to 5 Thesis. Minimum of three hours to be counted toward a master's degree. Prerequisite: consent of department.
601-1 per semester Continuing Enrollment. For those graduate students who have not finished their degree programs and who are in the process of working on their dissertation, thesis, or research paper. The student must have completed a minimum of 24 hours of dissertation research, or the minimum thesis, or research hours before being eligible to register for this course. Concurrent enrollment in any other course is not permitted. Graded *S/U* or *DEF* only.