Doctor of Philosophy in Computer Science

Background/Rationale

The program aims to develop computer scientists who are armed with methods, tools and techniques from both theoretical and systems aspects of computing. They should be able to formulate computing problems and develop new and innovative technology as novel solutions to address those problems. The graduates will gain expertise to independently contribute in Research and Development (R&D) on a specialized area of Computer Science. The program will prepare graduates for professional and/or research careers in industry, government or academe.

Requirements

1. Applicants must have a master's degree in Computer Science, Electrical and Electronics Engineering, Mathematics or other fields, provided they have a substantial background in computing. This must include the following:
   a. Data Structures
   b. Automata Theory
   c. Discrete Mathematical Structures
   d. Algorithms
   e. Computer Organization
   f. Operating Systems
   g. Computer Networks
   h. Database Systems
   i. Software Engineering
2. Applicants must demonstrate proficiency in at least one (1) high-level programming language prescribed by the department's graduate committee.
3. Applicants must get their endorsement by one (1) of the department's six (6) research laboratories:
   a. Algorithms and Complexity Laboratory (ACL)
   b. Computer Security Laboratory (CSL)
   c. Computer Vision and Machine Intelligence Group (CVMIG)
   d. Networks and Distributed Systems Group (NDSG)
   e. Scientific Computing Laboratory (SCL)
   f. Web Science Laboratory (WSL)

Curriculum

Theory Core Courses		3-6 units
Systems Core Courses		3-6 units
	Total:	9 units

In addition, students need to take the following:

Specialization Courses		9 units
Graduate Electives		6 units
CS 296 Graduate Seminar		1 unit
CS 400 Dissertation		12 units
	Total:	28 units

Total number of units for Ph.D.	37 units

Core Courses in Theory:
CS 204 (Theory of Computation)
CS 210 (Advanced Algorithms and Data Structures)
CS 214 (Parallel Algorithms)
CS 231 (Numerical Computing)
CS 360 (Formal Methods)

Core Courses in Systems:
CS 250 (Advanced Operating Systems)
CS 253 (Computer Security)
CS 255 (Advanced Computer Networks)
CS 260 (Advanced Software Engineering)
CS 268 (Web Science)
CS 270 (Advanced Database Systems)
CS 280 (Intelligent Systems)
CS 351 (Pervasive Computing and Communication)
CS 369 (Semantic Web)

The additional Specialization Courses should be taken from either Theory or Systems. These courses will depend on the particular research laboratory and research area that the student wishes to go into and will be decided by the graduate committee upon endorsement by the student's program adviser.

Specialization Courses:

Theory:
CS 204 Theory of Computation
CS 208 Complexity Theory
CS 210 Advanced Algorithms and Data Structures
CS 211 Combinatorial Optimization
CS 213 Communication Theory
CS 214 Parallel Algorithms
CS 216 Randomized Algorithms
CS 222 Programming Language Theory
CS 231 Numerical Computing
CS 236 Scientific Computing
CS 237 Biomedical Informatics
CS 239 Parallel Computing
CS 240 Computer Graphics
CS 242 Data Visualization
CS 247 Cryptography
CS 271 Database Theory
CS 290 Advanced Topics in Theoretical Computer Science
CS 294 Advanced Topics in Computational Science
CS 301 Communication Complexity
CS 315 Algorithms in Bioinformatics
CS 318 Approximation Algorithm
CS 334 Computational Systems Biology
CS 338 Computational Models of Biological Processes
CS 360 Formal Methods
CS 390 Independent Study
CS 397 Special Topics
ES 201 Advanced Mathematical Methods in Engineering I
ES 202 Advanced Mathematical Methods in Engineering II

Systems:
CS 220 Survey of Programming Languages
CS 250 Advanced Operating Systems
CS 253 Computer Security
CS 255 Advanced Computer Networks
CS 256 Computer Systems Performance Analysis
CS 257 Distributed Systems
CS 258 Mobile Computing
CS 259 Network Performance, Modeling and Monitoring
CS 260 Advanced Software Engineering
CS 262 Methods of Software Development
CS 265 Software Quality Assurance
CS 266 IT Project Management
CS 267 Software Engineering for the Web
CS 268 Web Science
CS 270 Advanced Database Systems
CS 280 Intelligent Systems
CS 281 Robotic Systems
CS 282 Computer Vision
CS 283 Data Mining
CS 284 Machine Learning
CS 286 Natural Language Understanding
CS 291 Advanced Topics in Net-Centric Computing
CS 292 Advanced Topics in Software Technology
CS 293 Advanced Topics in Computer Systems
CS 295 Advanced Topics in Intelligent Systems
CS 341 Knowledge Engineering
CS 351 Pervasive Computing and Communication
CS 369 Semantic Web
CS 360 Formal Methods
CS 371 Security Engineering
CS 380 Computational Intelligence I
CS 381 Computational Intelligence II
CS 385 Artificial Neural Networks
CS 389 Digital Image Processing
CS 390 Independent Study
CS 297 Special Topics

Doctor of Philosophy in Computer Science

Course Description

Course No.	Course Name	Course Description	Prerequisite/ Corequisite	Credits
CS 204	Theory of Computation	Formal models of computation; recursive function theory; undecidability. Resource-bounded computational complexity, non-determinism, NP-completeness.	Pre: CS 133 or COI	3 u.
CS 208	Complexity Theory	Computational models, measures of complexity, complexity classes: nondeterministic, alternating, probabilistic, parallel. Boolean circuits. Complete problems.	Pre: CS 204	3 u.
CS 210	Advanced Algorithms and Data Structures	Advanced data structures: algorithm design techniques; mathematical techniques in the analysis of algorithms.	Pre: CS 135	3 u.
CS 211	Combinatorial Optimization	Design and analysis of algorithms for combinatorial optimization problems, worst-case complexity, NP-Completeness proofs, heuristics. Open problems.	Pre: COI	3 u.
CS 213	Communications Theory	Mathematical theiry of communication. Information Theory. Communication Channels. Coding. Cryptography.	Pre: COI	3 u.
CS 214	Parallel Algorithms	Models of parallel computation. Performance measures, scalability, pipelining, techniques for analyzing parallel algorithms. Interconnection network topologies. Applications.	Pre: COI	3 u.
CS 216	Randomized Algorithms	Construction and analysis of randomized algorithms. Expected performance of randomized algorithms, fundamental limitations on probabilistic computations, complexity issues, applications.	Pre: CS 135, Stat 112 or COI	3 u.
CS 220	Survey of Programming Languages	Comparative study of different types of modern programming languages: imperative, functional, logic-based and object-oriented. Syntax, semantics and implementation of programming languages.	Pre: CS 150 or equiv.	3 u.
CS 222	Programming Language Theory	Fundamental concepts underlying all programming languages. Semantic aspects including binding times, visibility, retention, storage management, abstraction mechanisms and extensibility. Operational and denotational semantic specifications.	Pre: CS 150 or equiv.	3 u.
CS 225	Compiler Design and Construction	Theory of compiler design and construction; techniques on error connection and recovery; code generation and optimization.	Pre: CS 220	3 u.
CS 231	Numerical Computing	Algorithm design for numerical computation. Error analysis. Performance evaluation of numerical software.	Pre: CS 131 or COI	3 u.
CS 236	Scientific Computing	Problems and methods in scientific computing. Applications from science and engineering.	Pre: COI	3 u.
CS 237	Biomedical Informatics	Computational methods for managing and analyzing information about biomedical systems. Standards and tools in Biomedical Informatics.	Pre: COI	3 u.
CS 239	Parallel Computing	Parallel computer architectures, Programming for parallel architectures. Representation, program dependence, control structures.	Pre: COI	3 u.
CS 240	Computer Graphics	Solid modelling: Euler operators, finite element methods. Rendering: filling, shading, ray tracing. Natural modelling: L-systems, fractals. Image processing: filtering, antialiasing, enhancement.	Pre: COI	3 u.
CS 242	Data Visualization	Visualization techniques for data from science, business, social science, demographics, and information management.	Pre: COI	3 u.
CS 247	Cryptography	Primality testing, finite fields, elliptic curves. Protocols: public key cryptography, digital signatures, zero-knowledge proofs, and other cryptographic protocols.	Pre: COI	3 u.
CS 250	Advanced Operating Systems	Synchronization and communication mechanisms; virtual memory management, file systems, deadlock control, resource allocation, protection and access control. Case study of specialized systems.	Pre: CS 140 or equiv	3 u.
CS 253	Computer Security	Encryption, digital signatures, authentication, key	Pre: COI	3 u.

		management. Secure electronic commerce. Network security. File security.		
CS 255	Advanced Computer Networks	The five-layer reference model: physical, data link, network, transport, application. Distributed computing. Networked multimedia systems. Client0server computing. Communication and internetworking.	Pre: CS 250	3 u.
CS 256	Computer Systems Performance Analysis	Overview of performance evaluation. Measurement techniques and tools. Applications of probability theory and techniques. Experimental design and analysis. Simulation and queuing models.	Pre: COI	3 u.
CS 257	Distributed Systems	Computer communications networks and their protocols. Event ordering and synchronization. Deadlocks. Network operating systems and languages for distributed computing. Distributed databases. Fault tolerance and recovery strategies. Applications.	Pre: CS 140 or equiv.	3 u.
CS 258	Mobile Computing	Mobile computing systems. Data management, packet transmission, mobile IP, routing protocols, reliability and issues in mobile wireless networks.	Pre: COI	3 u.
CS 259	Network Performance, Modeling and Monitoring	Network Performance Evaluation. Measurement Techniques and Tools. Simulation, queuing models, case studies, practicals.	Pre: CS 255, CS 256 or COI	3 u.
CS 260	Advanced Software Engineering	Structured approach to requirements analysis, system design, implementation and maintenance of software systems. Formal description and documentation techniques.	Pre: CS 192 or equiv.	3 u.
CS 262	Methods of Software Development	Modern approaches to software development. Prototyping and automated tools. Computer Aided Systems Engineering (CASE) methods and tools. Object-oriented Programming Systems (OOPS).	Pre: CS 260	3 u.
CS 265	Software Quality Assurance	Quality Management, Quality Assurance, Quality Control. Measurement and Analysis. Maturity Models.	Pre: CS 260 or COI	3 u.
CS 266	IT Project Management	Detailed discussions of project management knowledge areas and processes. Case studies. Simulations and walkthrough of real-world IT projects from initiation and planning to evaluation and closing.	Pre: COI	3 u.
CS 267	Software Engineering for the Web	Software processes and requirements analysis, design, development and testing for web-based systems, software development tools, configuration management systems, case studies.	Pre: CS 260, CS 270 or COI	3 u.
CS 268	Web Science	The Web as a full communications medium that foster full collaboration, social interaction and commerce. Case studies.	Pre: CS 267	3 u.
CS 270	Advanced Database Systems	Data models and their underlying mathematical foundations; database manipulation and query languages; functional dependencies; physical data organization and indexing methods; concurrency control; crash recovery; database security; distributed databases.	Pre: CS 250	3 u.
CS 271	Database Theory	Relational database model, query languages, domain independence, relational calculus, query optimization, constraints.	Pre: CS 165 or equiv.	3 u.
CS 280	Intelligent Systems	Fundamental issues in Intelligent Systems. Intelligent search and optimization methods. Knowledge representation and reasoning. Learning, natural language understanding, pattern recognition, knowledge-based systems and other methods in intelligent systems.	Pre: COI	3 u.
CS 281	Robotic Systems	Biologically-motivated robotic systems. Reactive, deliberative, and hybrid architectures. Knowledge representation for robotic systems. Sensor fusion and perceptual strategies. Adaptation and Social behavior.	Pre: CS 280 or equiv.	3 u.
CS 282	Computer Vision	Image formation. Early vision. Segmentation from texture and motion. Object representation. Matching and Inference. Knowledge-based vision.	Pre: COI	3 u.
CS 283	Data Mining	Decision trees, association rules, clustering. Intrusion detection. Design and use of serial, distributed and parallel data mining algorithms.	Pre: CS 165, CS 280 or COI	3 u.
CS 284	Machine Learning	Pattern recognition, parametric and non-parametric	Pre: CS 280 or	3 u.

		learning, decision trees, Bayesian and neural networks, reinforcement learning, genetic algorithms, computational learning theory.	COI	
CS 286	Natural Language Understanding	Computational properties if natural languages. Morphological, syntactic and semantic processing from an algorithmic perspective. Models of acquision and parsing.	Pre: CS 280 or COI	3 u.
CS 290*	Advanced Topics in Theoretical Computer Science	-	Pre: COI	3 u.
CS 291*	Advanced Topics in Net-Centric Computing	-	Pre: COI	3 u.
CS 292*	Advanced Topics in Software Technology	-	Pre: COI	3 u.
CS 293*	Advanced Topics in Computer Systems	-	Pre: COI	3 u.
CS 294*	Advanced Topics in Computational Science	-	Pre: COI	3 u.
CS 295*	Advanced Topics in Intelligent Systems	-	Pre: COI	3 u.
CS 301	Communication Complexity	Communication complexity and its applications to parallel computing. Communication complexity measures, lower bound techniques.	Pre: CS 210 or COI	3 u.
CS 315	Algorithms in Bioinformatics	Algorithm design techniques applied to Bioinformatics Problems, DNA Sequencing, Gene Expression Analysis, and Protein Folding.	Pre: CS 135 or COI	3 u.
CS 318	Approximation Algorithms	Approximate methods for solving a wide range of intractable of hard problems.	Pre: CS 210 or COI	3 u.
CS 334	Computational Systems in Biology	Analysis of molecular interaction networks and pathways. Qualitative and quantitative methods and tools applied to biochemical systems.	Pre: COI	3 u.
CS 338	Computational Models of Biological Processes	Formal models and heuristic approaches for biological processes.	Pre: COI	3 u.
CS 341	Knowledge Engineering	Reasoning systems to support decision making, learning and action.	Pre: CS 280 or COI	3 u.
CS 351	Pervasive Computing and Communication	Technologies for ubiquitous computing.	Pre: CS 255, CS 267, CS 280 or COI	3 u.
CS 360	Formal Methods	Program specification, verification, and refinement.	Pre: COI	3 u.
CS 369	The Semantic Web	Web technologies, ontology engineering, social network analysis and web standards.	Pre: CS 267	3 u.
CS 371	Security Engineering	Security issues in the design and engineering of software-based systems for different platforms.	Pre: CS 253 or COI	3 u.
CS 380	Computational Intelligence I	Metaheuristic algorithms and their utility in solving engineering and scientific problems.	Pre: CS 280 or COI	3 u.
CS 381	Computational Intelligence II	Agent-based systems and their applications to real-world problems.	Pre: CS 280 or COI	3 u.
CS 385	Artificial Neural Networks	Computational models inspired by the structural and functional aspects of the brain.	Pre: CS 280 or COI	3 u.
CS 389	Digital Image Processing	Tools and techniques for transformation of images for subsequent human or machine interpretation.	Pre: COI	3 u.
CS 390[1]	Independent Study	-	Pre: completion of 18 u. of CS graduate courses including 6 units of Specialization courses	3 u.
CS 397[2]	Special Topics	-	Pre: COI	3 u.
CS 400[3]	Dissertation	-	Pre: completion of all coursework requirements for the program	12 u.

* may be repeated for a maximum of 6 u.; topic should be indicated for record purposes
1 may be taken twice
2 may be taken twice provided that topics are different
3 may be spread over two (2) semesters
COI: Consent of Instructor