COMPUTER SCIENCE (Ph.D.)

Executive Officer: Distinguished Professor Robert Haralick
The Graduate Center
365 Fifth Avenue
New York, NY 10016
Email: Compsci@gc.cuny.edu
http://www.cs.gc.cuny.edu

FACULTY

THE PROGRAM
The Ph.D. Program in Computer Science is designed to prepare selected students for leadership in industrial careers and research as well as in teaching and academic research. The ubiquitous role of the computer in our society requires that the Ph.D. candidate master the discipline of computer science in its broadest sense as well as display knowledge of a specialized area and perform independent research.

Areas of Study
The program is particularly strong in the following specializations. (Please note that the division into areas of study is somewhat artificial; some courses are relevant to more than one area, depending on the instructor’s focus, could be placed in another area.)

Programming Languages and Software Methodologies
Programming language development has been an active area of research in computer science almost from the origin of computer science itself. Nowadays, programming languages are defined formally. Stylistically, a programming language can be classified either as an imperative language or as a declarative language. Programs written by a user in a particular programming language should make use of computer software development methodologies. These methodologies not only foster good or correct practice in writing a program, but include techniques that cover the range of phased activities that a software product goes through from its conception through implementation to its maintenance. “Software Engineering” techniques are included in this category. Current faculty interests include formal methods of program description, verifying program correctness, declarative language construction, and mathematical linguistics.
Theoretical Computer Science and Its Applications

Predating the field of computer science, theoretical computer science is a mathematically rigorous study of computing. It includes a theory of computing machines, solvability, formal language theory, and concepts of timing. The area is so basic that it is often called “foundations.” Topics include formal languages, automata theory, computability and unsolvability, and logic of programs. Current faculty interests include computational geometry, security, recursion theory, applied logic, and computational complexity.

Artificial Intelligence, Cognitive Science, and Adaptive Systems

Artificial intelligence (AI) and cognitive science are concerned with developing algorithmic methodologies that can mimic various aspects of human performance and their implementation as computer programs. These methodologies include symbolic knowledge representation, concepts and methods of inference, modeling human thought and sensory-motor performance. Cognitive science includes developing methodologies that model neural systems and adaptive dynamical systems. Current faculty interests include computational linguistics, data mining, natural language processing, learning and understanding systems, human locomotion and balance control, neural networks, logic in artificial intelligence, including logic programming, knowledge and belief, and image recognition systems.

Scientific Computing and Modeling of Systems

The original impetus for the creation of a computing machine was the need to do large-scale numerical computations. The field of numerical computation techniques continues to grow, with numerical calculations still playing an important role in scientific research. New approaches and techniques evolve that are quite general and powerful. Simulation of systems likewise plays an important role in scientific inquiry and more broadly in the design of all systems (including computer systems). Analytic modeling is another tool useful for analyzing the behavior of designed but not yet implemented systems. Current faculty interests include simulation of continuous and discrete systems, statistical modeling of systems, numerical algebra, numerical analysis, and biomedical computing.

Algorithms and Their Analysis

Algorithm design is at the heart of computing. Algorithms are the detailed procedures that in a finite number of steps accomplish a computing task. Thus this is a broadly defined category that impinges on all other areas. Current faculty interests include cryptography, combinatorial algorithm design, run time complexity, parallel and distributed algorithm design, and analysis of algorithms.

Computer Architecture, Networks, and Communications Systems

With the dynamic development of computer technology, hardware and computer architecture are important areas of research and development. The courses offered in this area include advanced computer architecture and computer/network communications. Current faculty research includes computer networks, parallel computation, neural nets, petri nets, and telecommunications.

Media Processing, Computer Vision, and Graphics

The design, distribution, display, recognition, storage schemes, large data sets, and multiple media in a document are important applied research areas in computer science. Medical information processing is a closely aligned research area. It teams physicians and computer scientists and has the potential of producing significant health-related goals. CUNY has a number of faculty members interested in this area. Current interests include graphics, computer vision, document understanding, database technology and document storage and retrieval, medical information processing, digital topological techniques for image processing, real-time processing of biomedical signals, and multiresolution approaches for image understanding.

Courses in the Ph.D. Program in Computer Science are offered at the Graduate Center as well as at Baruch College, Brooklyn College, the City College, Queens College, and the College of Staten Island.
SPECIAL REQUIREMENTS FOR ADMISSION

In addition to the University’s requirements for admission stipulated earlier in this bulletin, the applicant is expected to have attained a minimum average of B in his/her undergraduate major and to have completed course work equivalent to an undergraduate major in computer science. Exceptions will be considered by the program’s Admissions Committee for those applicants with an undergraduate major in one of the fields cognate to computer science or with extensive experience in the field.

Specifically, entering students are expected to have a background (minimally, at an undergraduate level) in the following areas: Operating Systems; Fundamental Algorithms; Object-Oriented Programming (e.g., C++ or Java); Databases; Discrete Mathematics; Computer Networks; Theoretical Computer Science (Logic, Models of Computation, Analysis of Algorithms); Programming Languages; and Probability.

Students who are admitted with deficiencies in their background will be required to take graduate (or undergraduate) courses to make up for them prior to attempting the core courses. Courses that are required to fulfill deficiencies can be included in the first 30 credits of the degree if they are approved, if they are graduate courses, and if the student achieves at least a B grade in the course.

SPECIAL REQUIREMENTS FOR THE DOCTOR OF PHILOSOPHY

The following requirements are in addition to the University requirements for the Ph.D. as stated earlier in this bulletin.

Program of Study
The curriculum for all doctoral students in the Ph.D. Program in Computer Science consists of a minimum of 60 graduate credits beyond the baccalaureate degree, in addition to undergraduate deficiencies that may have to be remedied. Transfer credits will be subject to the rules pertaining to CUNY doctoral students. The program offers core courses in three broad research areas as listed in the following:

Area 1: Algorithms and Theory
- CSc 70010 Algorithms (4)
- CSc 71010 Programming Language (3)
- CSc 73010 Cryptography and Computer Security (3)
- CSc 75100 Logical Fundamentals of Computer Science (3)

Area 2: Artificial Intelligence
- CSc 74011 Artificial Intelligence (3)
- CSc 74020 Machine Learning (3)
- CSc 74030 Computer Vision and Image Processing (3)
- CSc 74040 Natural Language Processing (3)

Area 3: Systems and Computational Science
- CSc 72010 Computer Networks (3)
- CSc 72020 Distributed Operating Systems (3)
- CSc 72030 Database Systems (3)
- CSc 76010 Parallel Scientific Computing (3)

First Examination
Students must complete the following requirements to pass the First Exam:
1. Pass CSc 70010, the Algorithms course, and pass its final written exam with a score of at least 70 points out of 100.
2. Pass four core courses, with at least one core course from each of the three research areas defined above, and with an average grade of B+ (GPA 3.3).

Students are required to finish above core course requirements by the end of the fourth semester.

Note: If a student takes more than four core courses, any of the additional core courses will be counted as electives.

Other course requirements: After successful completion of the First Examination, students must complete the following requirements to advance to candidacy:
1. Complete at least two 3-credit 80000 level courses, above 80020.
2. Complete CSc 80010, the Research Survey course.
3. Pass a Second Exam that has a written and an oral component. At the time of the Second Exam, the student must choose a mentor.
A Level II student has the option of taking the 4-credit course CSc 80020, Computer Science Research, for up to two times (over two semesters) with these credits being counted toward the 60 graduate credit requirement.

Second Examination After successful completion of the program’s First Examination requirement and completion of all required course work, the candidate must pass a two-part Second Examination. The Second Exam tests the student’s in-depth knowledge and understanding of a knowledge area directly related to his/her topic of dissertation research. The first part is a written survey study that is related to the student’s dissertation research, and the second part is an oral exam of the survey material in the format of a public seminar presentation. The Second Exam is judged by a student-chosen committee of at least three Computer Science doctoral faculty members, including the student’s mentor.

Research Tool Before advancing to candidacy, a student is required to show high-level programming proficiency. Students will satisfy this requirement by submitting to the Executive Officer a large computer program, written by themselves. The program must include relevant documentation. This program can be one written in industry, one developed on his/her own, or one developed as part of a course that requires the writing of a large program.

Dissertation Proposal Within two years of having advanced to candidacy, a student is expected to defend his/her dissertation proposal, which outlines the particular research project the student plans to undertake. This examination has a written and an oral part and is judged by a committee of at least three GC doctoral faculty members, including the student’s mentor and at least one other Computer Science doctoral faculty member.

Dissertation A student is required to complete a dissertation based on original research in one of the areas of specialization under the guidance of his/her faculty adviser and dissertation committee. The dissertation committee will consist of the student’s mentor and other members of the doctoral faculty whose areas of specialization are considered to be directly relevant to the student’s intended dissertation research topic. After the dissertation has been approved by the student’s mentor and the examination committee, the student must successfully defend it in an oral examination. The examination committee consists of the dissertation committee plus an outside member who is an expert in the field of the dissertation.

Courses
All courses are 3 credits, except as noted. Please note that some courses may be offered infrequently; consult with the program for further information.

Algorithms and Their Analysis
C SC 70010 Algorithms
4 credits
C SC 76010 Parallel Scientific Computing
C SC 80010 Algorithms for Parallel and Distributed Computation
C SC 80020 Topics in Combinatorial Algorithms
C SC 80030 Topics in Algorithm Design
C SC 80040 Topics in Algorithm Analysis
C SC 80200 Seminar in Algorithm Design and Analysis
1 credit

Programming Languages and Software Methodologies
C SC 71010 Programming Languages
C SC 72030 Database Systems
C SC 81010 Topics in Theoretical Underpinnings of Programming Language Design
C SC 81020 Topics in Computer Software Development
C SC 81030 Topics in Programming Languages
C SC 81040 Topics in Databases
C SC 81200 Seminar in Software Design
1 credit
Computer Architecture, Networks, and Communications Systems
C SC 72010 Computer Networks
C SC 72020 Distributed Operating Systems
C SC 82005 Advanced Computer Networks
C SC 82010 Computer Systems
C SC 82020 Computer Communication Systems
C SC 82100 Advanced Topics in Computer Systems
C SC 82110 Advanced Topics in Operating Systems
C SC 82200 Seminar in Computer Systems
1 credit

Media Processing, Computer Vision, and Graphics
C SC 74030 Computer Vision and Image Processing
C SC 83005 Computer Vision
C SC 83010 Topics in Computer Graphics
C SC 83020 Topics in Computer Vision
C SC 83030 Topics in Information Retrieval
C SC 83040 Topics in Document Analysis
C SC 83050 Topics in Image Processing
C SC 83060 Topics in Media Processing
C SC 83200 Seminar in Media Processing, Computer Vision or Graphics
1 credit

Artificial Intelligence, Cognitive Science, and Adaptive Systems
C SC 74011 Artificial Intelligence
C SC 74020 Machine Learning
C SC 74040 Natural Language Processing
C SC 84010 Topics in Artificial Intelligence
C SC 84020 Topics in Adaptive Systems
C SC 84030 Computational Models of Cognitive Systems
C SC 84200 Seminar in Artificial Intelligence and Cognitive Science
1 credit

Theoretical Computer Science and Its Applications
C SC 73010 Cryptography and Computer Security
C SC 75010 Theoretical Computer Science
C SC 75100 Logical Fundamentals of Computer Science
C SC 85010 Topics in Logics and their uses
C SC 85011 Logic in Computer Science
C SC 85020 Topics in Theoretical Computer Science
C SC 85030 Topics in Cryptography and Computer Security
C SC 85031 Discrete Mathematics for Cryptographic Applications
C SC 85040 Topics in Computational Complexity
C SC 85200 Seminar in Theoretical Computer Science
1 credit

Scientific Computing and Modeling of Systems
C SC 86005 Statistical Techniques and Probability Models in Computer Science
C SC 86010 Scientific Computing and Numerical Methods
C SC 86020 Probabilistic Modeling of Computer Systems
C SC 86030 Topics in Simulation Methodology
C SC 86200 Seminar in Scientific Computing
1 credit
C SC 86210 Seminar in Modeling Computer Systems
1 credit
Miscellaneous

C SC 79000 Independent Study/Research Project
C SC 79100 Research Survey
  1 credit
C SC 80000 Readings in Computer Science
C SC 80020 Computer Science Research
C SC 87100 Selected Topics in Computer Science
C SC 87200 Seminar General Topics
C SC 89000 Doctoral Dissertation Research
  1 to 6 credits
C SC 90000 Dissertation Supervision
  1 credit