CS PhD Program – Revised 5-09

Overview of the new requirements

The main requirements and milestones along the path of a PhD student will be:

- Background Knowledge Requirement (new);
- Breadth Course Requirement (revised);
- Preliminary Oral Exam (revised);
- Thesis Proposal Exam (new)

Background Knowledge Requirement

The concepts covered here are considered to constitute a minimal core body of knowledge with which all PhD graduates of our department should be familiar. These concepts are required prerequisites for many of our graduate classes; students must know these concepts to succeed in these classes.

Background concepts

- **Machine Architecture and Organization**. Covers basic hardware/software components of a computer system, including data representation, machine-level programs, instruction set architecture, processor organization, memory hierarchy, virtual memory, compiling, and linking.
- **Theoretical Foundations**. Must cover *one of* the following two bodies of knowledge:
    - **Algorithms and Data Structures** or **Formal Languages and Automata Theory**.
        - **Algorithms and Data Structures**. Analysis, data structures, and algorithms, e.g.: basic algorithm analysis (recurrences, asymptotic notation), basic data structures (lists, stacks, queues, heaps, hash tables, (balanced) binary search trees), basic algorithms (sorting, searching, graph traversal, shortest paths, minimum spanning trees).
        - **Formal Languages and Automata Theory**. Logical/mathematical foundations of computer science. Specific topics include formal languages, their correspondence to machine models, lexical analysis, string matching, parsing, decidability, undecidability, limits of computability, and computational complexity.
- **Operating Systems**. Topics include processes/threads, process coordination, interprocess communication, asynchronous events, memory management/file systems.
- **Programming & Software Development**. Topics include: design and analysis of programs, software development tools and methods, debugging, I/O, state machines, exception handling, testing, coding standards, software lifecycle models, requirements analysis.

Satisfying the Background Knowledge Requirement

The Background Knowledge Requirement may be satisfied in five different ways:

1. By passing the GRE Computer Science subject exam with a score in the 90th percentile or higher.

2. By passing an appropriate undergraduate course with a grade of B+ or higher. The appropriate courses at The University of Minnesota are noted below. However, a student may take such courses anywhere, and simply needs to point out on their transcript any qualifying courses. The Director of Graduate Studies is responsible for approving the use of courses to satisfy the background requirement. The relevant UMN courses are:

- **Machine Architecture and Organization =** CSCI 2021
- **Theoretical Foundations**:
  - **Algorithms and Data Structures =** CSCI 4041
  - **Formal Languages and Automata Theory =** CSCI 4011
- **Operating Systems** = CSCI 4061
- **Programming & Software Development =** CSCI 3081

3. By passing the final exam for the appropriate UMN class with a grade of B+ or higher

4. By passing a graduate course with a grade of B+ or higher for which an appropriate undergraduate course is a clearly defined prerequisite. For example, at the University of Minnesota, CSCI 5421 "Advanced Algorithms and Data Structures" has CSCI 4041 "Algorithms and Data Structures" as a prerequisite. Thus, getting a B+ in 5421 is evidence that a student has adequate background in Algorithms and Data Structures. Students must check with the Director of Graduate Studies to verify that a specific graduate course demonstrates knowledge of a particular background area.

5. By petitioning the Director of Graduate Studies to accept some other experience as evidence of adequate background. For example, a student could have extensive industrial software development experience without having taken a course on software development.

Students must satisfy the background requirement within their first year in the PhD program. If they are not able to do so, they may – with the support of their advisor – petition the Director of Graduate Studies for an extension.

Prerequisite Table – Graduate courses for which background undergraduate courses are substantial prerequisites

CSci 2021	CSci 4041	CSci 4011	CSci 4061	CSci 3081
5204	5403	5106	5103	5106
	5421	5161	5105	5161
	5461		5211	5801
	5471			
	5481			
	5523			

Breadth Course Requirement

The purpose of the Breadth Course Requirement is to expose students to diverse Computer Science research topics and methods. PhD student must take a total of five (5) courses that satisfy the following requirements:

- Each of the five courses is in a different **sub-area**.
- The student must take at least one course in each different breadth **area**.
- The student **must** take one course in the Theoretical Foundations sub-area.

PhD students must have an average GPA of 3.45 or higher for the five courses they use to satisfy the Breadth Course Requirement. Students have three (3) years to satisfy this requirement. If students want to take a more advanced course in a sub-area than the listed options – typically, one for which one of the listed options is a prerequisite – they may petition the Director of Graduate Studies to use this course for satisfying the requirement. Students may petition the Director of Graduate Studies to transfer credit for up to two courses to use for satisfying the Breadth Course Requirement.

Master's students (MS and MCS) are required to take three (3) courses, one from each of the areas. Students must maintain an overall GPA of 3.0 for MCS and 3.25 for MS candidates for all courses on their degree program, as well as those used to satisfy the breadth requirement. Substitutions are rarely permitted and transfer courses will not count towards the breadth requirement.

All courses must be taken for graduate credit and on the A-F grading basis.

Breadth Areas

There are three breadth areas:

- **Theory and Algorithms**
- **Architecture, Systems, and Software**
- **Applications**

Each area contains a number of sub-areas, and each sub-area contains a number of courses. Defining sub-areas within areas allows for clustering related courses and for

increasing the diversity of the courses a student will take to satisfy the breadth requirement.

Theory and Algorithms

1. Theoretical Foundations (*note: all students **must** take one course in this sub-area*)
    - 5421: Advanced Algorithms & Data Structures
    - 5403: Computational Complexity
    - 5304: Computational Aspects of Matrix Theory
2. Applied Algorithms
    - 5302: Analysis of Numerical Algorithms
    - 5471: Modern Cryptography
    - 5481: Computational Techniques for Genomics
    - 5525: Machine Learning

Architecture, Systems, and Software

1. Programming, Software, Languages, Compilers
    - 5106: Programming Languages
    - 5161: Introduction to Compilers
    - 5801: Software Engineering I
2. Systems Software
    - 5103: Operating Systems
    - 5105: Foundations of Modern Operating Systems
    - 5451: Introduction to Parallel Computing: Architectures, Algorithms, and Programming
    - 5708: Architecture and Implementation of Database Management Systems
3. Architecture
    - 5204: Advanced Computer Architecture
4. Networking
    - 5211: Data Communications and Computer Networks
    - 5221: Foundations of Advanced Networking
    - 5231: Wireless and Sensor Networks

Applications

1. Intelligent Systems: AI, Robotics, Machine Learning, Vision
    - 5511: Artificial Intelligence
    - 5521: Pattern Recognition
    - 5551: Introduction to Intelligent Robotic Systems
    - 5561: Computer Vision
2. Data Mining and Bioinformatics
    - 5523: Introduction to Data Mining
    - 5461: Functional Genomics, Systems Biology, and Bioinformatics
3. Graphics, Visualization, Human-Computer Interaction, Social Computing
    - 5107: Fundamentals of Computer Graphics 1

- 5109: Visualization
- 5115: User Interface Design, Implementation and Evaluation
- 5125: Collaborative and Social Computing

4. Security
    - 5271: Introduction to Computer Security

Preliminary Oral Examination

The goal of the Preliminary Oral Examination is to serve as an early test of a student's research abilities. It is **not a thesis proposal**: a separate examination is being added to serve that purpose.

Timing issues

The Oral Prelim should be taken as soon as the student is ready. Students **must take** the exam no later than their second year in the PhD program (however, with the support of their advisor, students may petition the Director of Graduate Studies for an extension). Students **must pass** the exam by the end of their third year. Before scheduling the Oral Prelim, students must submit a written report to their examining committee. Once the committee approves that report, the student may schedule the Oral Prelim. Therefore, students should submit their written report to their examining committee at least two months before their preferred Oral Prelim date.

Best Practices

The written report and oral exam must demonstrate the student's ability to do research. Different areas and different advisors use different methods and have different expectations of what a student must do to demonstrate research ability. Therefore, we cannot specify precisely what a student's report and oral presentation must include. However, we can offer a few “best practice” examples that would be acceptable in most or all research areas.

- **Completed research project**. A report on a research project **completed while a graduate student at the University of Minnesota**. Ideally, this project will have resulted in a published paper where the student is a major contributor.
- **Literature review**. A careful and insightful review of research in the student's specialty. This review should demonstrate a student's understanding of key research topics and methods in the area and show that he or she can identify interesting open research problems and appropriate means to address those problems.

Written Report

The report should be at least the length of a published conference paper, say 6000-8000 words, or 8-10 pages in the ACM SIG Proceedings format (http://www.acm.org/sigs/publications/proceedings-templates).
The same committee will examine both the written report and the oral exam.

Exam Scope and Format

The student will present the material in the written report. The committee will question the student about that material and directly related material, such as the methods that were used and possible alternative methods, ideas for future work, potential problems and obstacles. The committee is encouraged to probe the student's understanding of *related* material and concepts.

Possible outcomes of the exam

The committee may pass the student or fail the student. If the student fails, the committee may or may not choose to give the student another chance to pass the exam. A student can have at most two chances to pass the Oral Prelim. As stated above, students must pass the exam within three years of entry to the PhD program.

Committee Composition

The Graduate School requires that the committee include three members from the Computer Science graduate faculty and one external member. The student in consultation with her/his advisor, will nominate three members, two internal (including the student's advisor) and one external. The DGS will approve these choices and will appoint one member of the committee from the department's Preliminary Oral Examination committee, selecting a person who is not in the student's research area.

Reminder: the same committee will examine both the written report and the oral exam.

Relationship to a student's M.S. research (Plan A Thesis / Plan B Report)

The same piece of research can be used to satisfy both the MS (Plan A or B) and Oral Prelim requirements. If this is done, there are four possible outcomes of the exam:

- The student can pass both exams.
- The student can fail both exams.
- The student can pass the MS, fail the Oral Prelim, and be given the option to retake the Oral Prelim.
- The student can pass the MS, fail the Oral Prelim, and not be given the option to retake the Oral Prelim.

.

Thesis Proposal Examination

The Oral Prelim will no longer cover the thesis proposal, and thus a Thesis Proposal Examination will also be required. Students will schedule a thesis proposal examination when their advisors believe they are ready. This is designed to speed up students' research progress. We expect that some students will be ready to do a thesis proposal In their third year in the program and that most will do it by their fourth year. The Graduate School requires that the dissertation be completed and defended within five years from the semester in which the student passes the prelim oral. Thus the thesis proposal examination should be passed within a reasonable time frame from the date of the prelim oral. The student, in consultation with her/his advisor, will choose the three internal (CS) and one outside department members which will be approved by the DGS.

Transition Plan

To detail how the new program will be phased in, we divide PhD students into three cohorts: those who are entering the program in Fall 2009, those who entered in Fall 2008, and those who entered before. We detail how each of them will be affected by the new degree requirements.

- **Fall 2009 cohort**. They must satisfy the new requirements.
- **Fall 2008 cohort**. They may choose whether to satisfy the new requirements or the old requirements (the requirements at the time they were admitted). If they choose to satisfy the new requirements, they may petition the DGS for an extension in the time they have to do their Preliminary Oral Examination (if necessary).
- **Previous cohorts**. They must satisfy the old requirements. This is necessary because all of these students have (or should have) taken the WPE.