University of Wisconsin-Madison
Computer Sciences Department
Assessment Plan
Spring 2013

PROGRAM OVERVIEW

Initially started in 1963 as the Department of Numerical Analysis, UW–Madison is home to one of the oldest computer science programs in the country. The Department of Computer Science in the College of Letters and Sciences offers B.S. and B.A. degrees, as well as an M.S. and Ph.D. program. Our curriculum includes project-oriented courses in computer animation, computer architecture, databases, networking, operating systems, programming languages and compilers, and software engineering; these systems courses are complimented by analysis courses in algorithms and complexity, numerical analysis, and math programming.

GOALS OF THE UNDERGRADUATE AND GRADUATE PROGRAMS

The mission of the Computer Science Department's **undergraduate** program is to prepare students for CS jobs or graduate studies in Computer Science. Fulfilling this mission involves ensuring that our majors receive a rigorous and broad Computer Science education that will make them attractive to prospective employers and to graduate schools.

As minimal learning objectives, students receiving their B.A. or B.S. degrees are expected to know the following:

- Students can understand and create robust, user-friendly, well-structured and well-documented programs in current programming languages such as Java and/or C.
- Students can implement and use appropriate data structures (e.g., stacks, queues, trees, graphs, and hash tables)
- Students can develop, implement, and analyze different algorithms (e.g., sorting and searching)
- Students have a basic understanding of how modern computers work in both hardware (e.g., computer architecture) and software (e.g., operating systems, compilers, or networking)
- Students can develop, implement, and analyze complex applications for different domains (e.g., in the fields of graphics, artificial intelligence, human/computer interaction, numerical analysis, or optimization)

The goals of the department's **graduate** program include preparing Masters students for successful CS careers, and preparing Doctoral students for successful research and teaching positions, either in industry or academia.

In addition to these academic goals, the Computer Sciences Department is committed to increasing the representation of women and minorities in Computer Science through outreach programs designed to attract under-represented groups to Computer Science (at the undergraduate level) and special efforts to ensure that women and other under-represented groups are encouraged and supported.

STRATEGIES FOR MEASURING STUDENTS' PERFORMANCE

Assessment and re-evaluation of our academic programs at both the undergraduate and graduate levels have included the following activities.

1. At the end of each semester, students evaluate each course and each instructor with an in-class anonymous survey.
2. Faculty continually evaluate the courses they teach, and update them to reflect the state-of-the-art in our rapidly evolving field.
3. As the field of Computer Science evolves and expands, holes in our programs are identified and appropriate new courses are introduced.
4. Faculty maintain ties to both local and national companies and research labs; contacts at these places provide regular feedback on their perceptions of the quality and level of preparedness of our students.
5. An active Course and Curriculum Committee regularly reviews our curriculum and proposes changes. The committee also works closely with other departments (e.g., in the College of Engineering) to update courses required by both Engineering and Computer Sciences.
6. The Associate Chair works closely with our Faculty Associates to discuss curriculum issues in our introductory courses, and to provide them with regular feedback on their teaching.
7. Teaching evaluations of Assistant Professors are performed by senior faculty as part of the annual-review process.
8. Surveys are distributed to undergraduate majors at department-sponsored lunches each spring to assess perceptions about climate issues.
9. Students are asked to fill out exit surveys that provide information about the jobs they have accepted and about their experience in our department.

All of these assessment steps have been followed regularly with the exception of an exit survey. Therefore, we plan to further investigate the data that would be most useful for us to collect.

TIMELINE

In addition to the on-going assessment activities listed above, Computer Sciences is currently (or planning to) investigating the following specific issues for our undergraduate majors:

A) Do students adequately learn the basics of how computers work and how to perform systems programming (e.g., in languages such as C) in our introductory course CS 354? Specifically, do students meet the following learning objectives:
   - Students can read and understand complex sequences of assembly code (e.g., x86)
   - Students can read, understand, and generate serious projects in the C programming language
   - Students can use a compiler, debugger, and disassembler (e.g., gcc toolchain)

    - Students can use a real code editor (e.g., emacs or vim)

B) Are students prepared for a career related to the field of computer science after obtaining a B.A. or B.S. degree?

C) Should all CS students be required to take the three introductory courses of CS 302, CS 367, and CS 240 and obtain a minimum GPA before they can declare a CS major?

To address these three issues, we propose the following multi-year plan implemented by the Associate Chair and Curriculum Committee in consultation with the CS faculty:

Pre-2012-2013:

1. Identified issues 1, 2 and 3 above for further attention
2. Surveyed CS majors in a selection of upper-level CS courses about required introductory courses (i.e., their usefulness, difficulty, like-ability, and need)

2012-2013:

1. Identified need for exit survey of CS undergraduates and graduates
2. Introduced Pilot Section of CS 354 taught by a faculty member of the curriculum department. The pilot course covered topics to emphasize systems programming and the instructor assessed student reactions to each new aspect of the course. (Issue A)
3. Surveyed CS majors at a department-sponsored lunch about whether or not they feel prepared for a CS career (Issue B)
4. Surveyed CS majors at department-sponsored lunch about perceptions of introductory courses required to declare CS major (CS 302, 367, and 240)
   (Issue C)
5. Analyzed grade correlation between introductory and upper-level courses
   (Issue C)

2013-2014 Plan:

1. Perform exit survey for graduating students; analyze results and identify
2. Implement suggested changes from pilot section of CS 354 into all sections as taught by Senior Lecturer (Issue A)
3. Develop additional 1-credit courses focusing on practical technical and other programming skills (Issue B)
4. Analyze and consider potential changes to requirements for declaring CS major (Issue C)

2014-2015 Plan:

1. Evaluate effectiveness of changes to CS 354 (Issue A)
2. Introduce new 1-credit courses for additional programming skills (Issue B)
3. Potentially implement changes to requirements for declaring CS (Issue C)