Assessment Term Report for

Winter 2004

by Robert Kerbs

Summary

Progress was made in refining the Computer Science Department's accreditation procedures – ensuring continual program improvement. The accreditation web site was enhanced so that full-time and part-time faculty alike would be better appraised of accreditation activities and timelines. A secured filing system was implemented so that faculty would no longer need to commit time to blocking out student names in their assessment portfolios – thus reducing the paperwork burden. The curriculum committee took up a number of issues identified from last term. Graduating Student Surveys were conducted to ensure we understand student concerns. It is planned that in Spring 2004 the department's Advisory Board will convene to determine current industry requirements of our students.

Assessment Activities:

The following activities were completed this term:

1. Computer Science Accreditation web site – modified to include *Advisory Board Meeting Minutes* and the *Graduating Student Survey Form*.
2. *Learning Outcomes* were created for CS 380 (Introduction to Computer Networks). This was necessary as this course is now part of the core for students now entering the Computer Science Program.
3. A new locking filing cabinet was procured and installed in the Computer Science Department's Conference Room.
4. The Curriculum Committee took up a number of issues identified in Fall 2003.
5. Portfolios were collected and analyzed. Please see below for details.
6. Graduating Student Surveys were collected and analyzed. Please see below for details.

Portfolio Findings

1. An instructor utilized WebCT to post select materials to the web and found that students were more successful with this material than in the past.
2. It was acknowledged that there is a lot of material to cover in CS 130 (Discrete Structures) and that consideration should be given to the following solutions: break CS 130 into two course *OR* offer an honors version of CS 130.

3. Students are generally responding positively to group assignments in a number of courses. It is felt that the stronger students are helping others to better understand important concepts.
4. There seems to be a continuing English comprehension problem in many classes. In addition, students do not seem to read their books, instead they rely on instructor provided notes and lectures.
5. One instructor found it useful to assign programming homework inclass and then have students present their results to the rest of the class.
6. An instructor required students to keep their own portfolios – worth 15% of their grade. If constructed properly, students could use the folios as a useful culmination of their experience in a given course. Each portfolio had four parts: syllabus and notes, set of algorithm descriptions, all homework turned in as well as quizzes, and corrected midterm and projects.
7. MATLAB was utilized in one section of CS 301 (Numerical Methods) for inclass demonstrations. It was found that students were more successful in implementing course projects than in the past.

Graduating Student Survey Findings

Here is what students are telling us as far as areas needing improvement:

1. Professors can be too demanding and difficult to understand. They could be more helpful and enthusiastic.
2. Need more practical projects.
3. Need more variety in programming languages.
4. It can be hard to get classes. Would like more professors and more classes.

Advisory Board Findings

We intended to have meetings with our Advisory Board this term but it was felt that we would have a better turnout if we had them next term (Spring, 2004).

New Recommendations

Based on findings from student portfolios, discussion with faculty, and *Graduating Student Surveys*, the following possible changes/actions should be taken or explored:

1. The Sun laboratory's future configuration should be looked at and plans for refreshing the equipment should also be made.
2. The frequency with which we assess our curriculum should be revisited. The goal is to obtain meaningful data with the least paperwork burden.
3. Consider offering an honors version of CS 130 (Discrete Structures).
4. Consider utilizing WebCT in a wider selection of courses.
5. Consider assigning inclass programming assignments where students present their solutions to the rest of the class.
6. Expand the use of student created portfolios in other classes.
7. Instructors should try to assign more practical programming assignments.

Prior Recommendations in Progress

[2003 Fall] Department should consider requiring an entry-level examination before students are admitted into the computer science program.

[2003 Fall] There should be the capability for students to join informal study groups.
[2003 Fall] Objects first versus imperative programming first. This issue should be revisited for our core-course sequence.
Status – this issue has been forwarded to the curriculum committee for analysis.
[2003 Winter] Program Outcomes — while comparing our Program Outcomes against portfolio materials Program Outcome one seemed too broad for careful auditing. This outcome should be revisited for analysis.
[2003 Winter] Program Outcomes must be tuned so that auditing is straightforward.
[2003 Winter] More instructors should require an English-language preface to each submitted program.
[2003 Winter] Oral and written communication standards should be enforced in CS courses. One method to accomplish this would be through the assignment of additional writing assignments. For example, we could require additional essays, weekly journal entries, homework and exams that include essay writing, or projects that require a written entry. Another approach might be to require more group projects and presentations. Faculty would need to monitor and coach student writing.
[2003 Winter] We should intervene as early as possible for students who exhibit writing deficiencies. Such students should be directed to the University's Writing Center for tutoring help.
[2003 Winter] More courses should require written communication. More exam questions should require paragraph or longer responses in English. More programming assignments should require prose descriptions of the code being submitted.
[2003 Winter] More courses should emphasize the software life-cycle, starting at the introductory level. Readability and maintainability should be as important as correctness.
[2003 Winter] Retool courses at all levels to increase communication, teamwork, and process requirements.
[2003 Winter] Refocus program around "software design & build" theme.
[2003 Winter] Introduce case studies in selected courses to teach students to solve systems-type problems.
Status – this issue has been passed on to the curriculum committee for analysis.
[2003 Winter] Reorganize core/elective coursework to ensure competency in two languages (currently Java and C++) upon graduation.
Status – this issue has been forwarded to the curriculum committee for analysis.
[2003 Winter] Encourage more students to seek co-op/internship opportunities in their junior and senior years.
[2003 Winter] Incorporate web-based search into courses involving term papers, presentations, or substantive projects.
[2003 Winter] Expose students to benefits of gaining certification in specialized areas.

Recommendations Completed

[2004 Winter] CS 380 *learning outcomes* were created.
[2004 Winter] CS 435 – the curriculum committee looked into the prospect of narrowing the range of topics covered in this class. It was decided that the instructor teaching

the course should decide which topics should be covered in more detail than others.

[2004 Winter] CS 210 – Instructors teaching this course were informed that students were coming out of this class without adequate knowledge/expertise in sequential circuit design. The instructors agreed to explicitly cover this material and ensure that adequate coverage is given.

[2004 Winter] Department-wide programming style-sheet. It was decided that this would be too restrictive. Consequently, it is recommended that instructors utilize some type of style sheet to help students learn to program in a comprehensible way.

[2004 Winter] Accreditation web site expanded to include *Advisory Board Meeting Minutes* and *Graduating Student Surveys*.

[2004 Winter] A locking filing cabinet to house student assessment documentation was procured and installed the department's conference room. Consequently, the paperwork burden was reduced to faculty members as they no longer need to block out student's names on collected materials.

[2003 Fall] Learning outcomes created for CS 431 and CS 463.

[2003 Fall] Faculty given option to utilize alternative sampling technique called ESP.

[2003 Fall] Accreditation web site created.

[2003 Spring] An Assessment Coordinator was selected for the coming year.

[2003 Winter] Identified need for faculty Assessment Coordinator with at least 4 WTU release per quarter.