

# Assessment Term Report for

Fall 2003

by Robert Kerbs

## Summary

Fall 2003 was the first term a *Faculty Assessment Coordinator* was utilized to help with assessment-related issues. Activities during this term included creation of the Computer Science Department's Accreditation Website, introduction of an additional sampling scheme for student portfolio collection, and establishment of learning outcomes for two courses. In terms of procedural requirements, full-time faculty now understands what is needed for collection and analysis of student-related materials while most of the part-time faculty does not. Consequently, further communication with part-time faculty will be made to ensure the appropriate materials are collected and analyzed. In Winter, 2004, it is anticipated that the Computer Science Department's *Advisory Board* will be refreshed and consulted.

## Assessment Activities:

The following activities were completed this term:

- 1) Creation of the Computer Science Accreditation website. This website can be thought of as a clearing-house for accreditation-related matters of interest to members of the Computer Science Department at Cal Poly. It now includes the following web pages:
  - a) Portfolio Materials Collection – informs faculty members of the procedure that the Computer Science Department has developed and utilizes to collect student materials and faculty feedback. As new ideas are explored, and existing methods improved, this web page will be updated accordingly.
  - b) Assessment by Quarter – identifies the faculty members assessing courses for the current term. It also includes faculty member responsibility lists for prior quarters. To simplify form distribution, this web page includes *Learning Outcomes* forms as well as *Student Assessment* forms – both of which are course specific.
  - c) CS Assessment Term Reports – a quarterly archive of assessment activities. It includes analyses relating to past and current assessment issues.
- 2) Sampling – prior to Fall 2003 faculty randomly selected approximately five students for portfolio materials collection. In several classes, the random sample was not indicative of class learning as a whole. Consequently, half-

way through the term faculty was given the option of either continuing with their random selection of five students for each course, or selecting a representative copy of Excellent, Satisfactory, and Poor student work for each assignment (this method is referred to as the *ESP Portfolio Materials Collection Method*). The faculty who chose the ESP approach found the portfolios to contain more meaningful material. In the future, the two options will be offered to our faculty.

- 3) *Learning Outcomes* were created for CS 431 (Operating Systems) and CS 463 (Undergraduate Seminar). Prior to Fall 2003, *Learning Outcomes* had not been articulated for both of these courses.

## **Portfolio Findings**

1. Although students have taken the required pre-requisites for a given class, their ability to utilize this prior knowledge seems to be lacking.
2. Part-time faculty members did not seem to understand what was required of them for student portfolio collection and analysis — only one instructor provided a course summary.
3. The sampling technique of randomly selecting five students for portfolio collection was not found to be useful in a number of courses.
4. In-class assignments were introduced in two courses with positive results.
5. In-class group assignments were introduced in two courses with positive results.
6. Providing lecture materials on the web prior to lectures resulted in additional time for discussion and critical thinking for a few instructors.
7. Some incoming students have major problems with English comprehension.
8. Some students have problems understanding and implementing linked-lists even though they have completed our core sequence of courses.
9. One instructor introduced informal outside-of-class study groups with very good results.

## **Student Survey Findings**

No Student Surveys were scheduled during the Fall 2003 quarter.

## **Advisory Board Findings**

No meetings were scheduled with the Advisory Board during the Fall 2003 quarter.

## **Findings - Other**

- Many discussions ensued at CS Department Meetings pertaining to the programming paradigm utilized in our core sequence of classes (CS 140, CS 141, CS 240, CS 241). Some faculty question the effectiveness of students learning object-oriented programming first versus a teaching paradigm teaching imperative programming first. This concern has been forwarded to the CS Curriculum Committee to be addressed.

## **New Recommendations**

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

1. The curriculum committee should:
  - a. Address the issue of object-oriented programming versus imperative programming for our core-course sequence.
  - b. Consider reorganizing our core/elective coursework to ensure competency in two languages (currently Java and C++) upon graduation
  - c. Consider creating a department-wide programming style-sheet.
  - d. Consider moving some lecture hours in CS 210 from combinatorial circuit design and analysis to sequential circuit design and analysis.
  - e. Consider narrowing the number of topics covered in CS 435.
2. The faculty should consider:
  - a. Using the ESP method of data collection.
  - b. Giving more in-class assignments.
  - c. Giving more group assignments.
  - d. Posting lecture materials on the web before class.
  - e. Requiring students who complete our core sequence of classes to possess the capability of implementing linked-lists from scratch.
  - f. Creating informal student study groups.
  - g. Informing students of the importance of mastering prerequisite material and apprised of the consequences of an inability to perform at the required level.
  - h. Direct students with English comprehension problems to the Writing Center.
3. The department should:
  - a. Consider requiring an entry-level exam to students wishing to enter the CS undergraduate program.
  - b. Create *Learning Outcomes* for CS 380 (Introduction to Computer Networks) – it is becoming a core course.
4. The assessment coordinator should:
  - a. Work with Pam Hester to ensure that part-time faculty follows the assessment procedures.
  - b. Work with the department to refresh the *Advisory Board* and coordinate the next *Advisory Board Meeting*.

### **Prior Recommendations in Progress**

[Winter, 2003] 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.

[Winter, 2003] Program Outcomes must be tuned so that auditing is straightforward.

[Winter, 2003] More instructors should require an English-language preface to each submitted program.

[Winter, 2003] Students should be encouraged to write more readable programs. One idea is to adopt a department-wide style sheet for Java.

Status – this issue has been forwarded to the curriculum committee for analysis.

[Winter, 2003] 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.
- [Winter, 2003] 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**

- [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.