

Experiences with CC2001 at a Small College

John F. Dooley
Knox College
Department of Computer Science
Galesburg, IL 61401 USA
+1.309.341.774
jdooley@knox.edu

ABSTRACT

In this poster, we describe our experiences over the past two years using the CC2001 small department model for our computer science major. We discuss issues with the number of required courses, the mathematics requirement, course prerequisites, our capstone experience, and the relative lack of computer science theory in the new curriculum.

Categories and Subject Descriptors

K.3.2 [Computer and Information Science Education]:
Curricular models, Computer Science Education, Curriculum.

General Terms

Standardization.

Keywords

Curriculum, curricular models, Computing Curricula 2001, CS major requirements, CC2001, discrete mathematics.

1. INTRODUCTION

Knox College is a highly selective liberal arts college of approximately 1350 students in West Central Illinois, USA. Knox has offered a computer science major since 1979. Since that time, the department has periodically reviewed and updated the major to conform to the latest set of curricular guidelines from the ACM. During the 2003-2004 academic year our department went through an evaluation of our computer science major and changed the major to conform to the small department model of CC2001 [2, page 50]. Our process for updating our major, the problems we encountered with CC2001, and our proposals for overcoming them are described in [3]. The updated major began to be offered in the fall of 2004.

Over the past two years we have retired two intermediate level courses from our curriculum and added three new ones. We have created some new electives and initiated a capstone experience. Our prerequisite structure has changed as has the recommended sequencing of our courses. Overall, we are pleased with the move

ACM Copyright notice goes here.

from our old CC91 based major to CC2001.

However, the migration to the new major requirements has not been without its problems. Adding an extra course at the intermediate level has complicated the order in which our students can take those classes. This has made it difficult for some students to take computer science electives when they are offered. We are concerned about the dearth of computer science theory in the new major. We are also displeased with how our capstone experiences have been implemented and are re-working that requirement. We've also discovered that the amount of time laid out in CC2001 for certain topics is woefully inadequate. This means we end up not covering some topics in the core as thoroughly as expected.

We have offered every introductory and intermediate level course at least twice since the advent of the renewed major in fall 2004. We have revised the number and content of the topics in most of those courses. As a department we have discussed each of the courses and how they relate to the vision of the department.

As we quoted from [1, pg 86] in our previous paper [3, pg 196], "While a liberal arts major may not involve a large number of computer science courses, the core areas of the discipline can be carefully and fully covered. This emphasis on the core of the discipline means that a liberal arts program will emphasize fundamental principles, with applications following from and building upon these ideas in a hierarchical fashion." The remainder of this poster describes problems we encountered in implementing our renewed major, our strategies for overcoming them, and our opinions about CC2001.

2. REFERENCES

- [1] Walker, H. and Schneider, G. M., *A Revised Model Curriculum for a Liberal Arts Degree in Computer Science*, CACM, 39:12, December, 1996, pgs 85-95.
- [2] Computing Curricula 2001: Computer Science: Report of The Joint Task Force on Computing Curricula, IEEE Computer Society and The Association for Computing Machinery, December, 2001. <http://www.acm.org/education/curricula.html>
- [3] Dooley, John F., *Moving to CC2001 at a Small College*, Proceedings of the 9th annual SIGCSE conference on Innovation and technology in computer science education, University of Leeds, England, 2004, pgs 196-198.