Updated Cooperative Learning Computer Lab Proposal

September 15, 2001: Submitted via the Office of Computing and Information Science for the *Faculty Innovation in Teaching Grants Program*, **http://www.cit.cornell.edu/atc/innovation/**

January 15, 2002: Updated and submitted via Campus Life

David I. Schwartz
Computer Science Department
Cornell University

Abstract

This proposal is modified from the version that I submitted to the *Faculty Innovation in Teaching Grants Program*, which accepted the proposal. Since September 15, a team composed of representatives from Campus Life, Academic Technology Services, the College of Engineering, and the Computer Science Department have been working towards clarifying the goals and issues associated with developing a cooperative computer lab. In particular, based on the recommendations of the design team, I have included portions in the modified proposal to show that the focus of the lab reaches beyond the introductory programming courses from which the idea was initially proposed. Pages 4–9 contain a majority of the updates, which consist of a discussion of current limitations, design modifications, and an extensive list of proposed uses for the lab. The current plan for the lab is shown on page 8.

Introduction

As an introductory course where students first experience computer science and engineering, CS100 offers a great opportunity for students to explore their career options. But, due to fear that arises from perceived competition with more experienced students, less experienced students either drop out or leave technical fields altogether, which hampers diversity. So, CS100 has begun to incorporate non-competitive, cooperative-learning labs from the Academic Excellence Workshop program in Engineering. Because of the large course enrollment and need for specialized computer labs, the College of Engineering and Computer Science Department are requesting space from Campus Life to build a cooperative learning computer laboratory in the North Campus at Cornell. Not only would such a lab help with CS100, but the entire university, especially in attracting and maintaining women and under-represented minorities in computer science and engineering.

This laboratory will do the following:

- Provide laboratory space for Academic Excellence Workshops in computer science and engineering.
- Foster skill development in the computational sciences with workshops in other courses.
- Provide general lab space for students and university efforts such as high-school programs, distance learning, and staff/faculty/student teaching development.
- Serve as a model to test cooperative-learning concepts with computers.

The lab's space, structure, and resources will support sixteen students split into cooperative learning groups of two to four students with two facilitators. The lab will ideally contain nine computers, eight for the students and one for the instructors. The types and arrangement of furniture will provide a flexible environment for groups to form and "roam" around to discuss and develop concepts on the computers.

Statement of Intent: Learning Issues

I. *What are the specific teaching and learning issues that I hope to address with my proposal? How do these issues affect student learning?*

Recent studies indicate that technical fields, such as engineering and computer science, continue to lack women and under-represented minorities [1]. For instance, the field of computer science struggles to attract women into the major [2]. One way to address the problem of under-enrollment involves investigating a student's first exposure to a technical major in the introductory courses. For their first "taste" of computer science, students typically start in the introductory programming class CS100 [3]. Besides first-year students enrolled in the Arts & Sciences and Engineering colleges, many other students from all colleges and levels, including staff and faculty, take the course. CS100 teaches how to solve problems using computer programs using two languages called Java and MATLAB.

Although the "surge" of interest in computer science appears unabated, we do not know how many students avoid CS100 due to fear of computers and perceived competition, lack of programming experience, and the "rumor mill" that spreads "tales of pain and woe." Consequently, those that must take the course sometimes do so with great trepidation. (At the beginning of each semester, I am always approached by a group of students who admit being afraid!) Very often the less experienced students feel disadvantaged from the students with past experience, especially those with C++ skills learned in high school. But not all students enter CS100 with fear, however. Students with previous experience or those taking the course past first year have the opposite problem of overconfidence. Such students sometimes do not realize the rigors of the course and drop out, or sometimes even fail because of poor study habits, which usually means cramming and little practice. So, whether through panic or overconfidence, many students find the course challenging and demanding.

The competition for higher grades drives some students to rely on more experienced peers to do the work to build higher homework grades, which is ultimately counterproductive. Since programming is a skill that must be practiced, individuals that allow partners to do a bulk of the programming tend to suffer because CS100 emphasizes exams for the course grades to assess individual performance. Of course, CS100 could feasibly remove homework assignments altogether, but that would also remove the initial motivation for practice. Without "watering down" the syllabus in CS100, we need to not only market the course better, such that we convert the external competition between students into internal, or individualized, competition. Individuals should develop programming skills without worrying about students who arrive in CS100 with a portion, or even all, of the required concepts already mastered.

The coverage of material causes another problem because of restrictions in following the course syllabus. The graded work generally follows strictly required material which must serve the broad range of students. More advanced work entices students but may "turn off" less developed students because of the complexity. Thus, skill development often replaces opportunities for more complex demonstrations in a field of study. Introductory courses should not just build skills, but hopefully give a glimpse of a field of study to help students choose career paths.

Currently, advisors recommend that students who fear computers, lack any experience, and feel unprepared for CS100 take a "pre-introductory" course called *COM S 99: Fundamentals of Computer Programming* (CS99) [3], Originally, CS99 was created to alleviate the problems that women and underrepresented minorities might feel in the hopes of boosting enrollment. Deliberately segregating these groups, however, is not an option. While CS99 has shown some success, issues of fear and competition with experienced peers still persist in CS100, so CS99 has not solved the problems at hand.

A positive development that will help reduce the competition is the switching from C++ to Java in the Computer Science AP Exam three years from now [4]. Until the public school system switches, CS100 will still face ranges of experience. Even with experience, students might continue to take CS100 as an "easy A," which fearful students already blame as unfair. Also,

high-school programs do not maintain the same standards, resulting in disparate performances from students who supposedly have equivalent experience. Again, overconfidence and bad habits also contribute when some experienced students witness their poor performance on the exams. The experienced students who successfully make the transition present a different challenge: boredom. The exceptional students could be future leaders in their fields and should thus be encouraged without exacerbating the perceived competition.

To reduce the fear and rivalry that persist in CS100 and still maintain high standards, the computer science department and the Academic Excellence Workshop (AEW) program in Engineering OR&IS [5] started running cooperative-learning laboratory sections in CS100 in Spring 2000 [6]. In cooperative learning, groups of students work together to solve problems in a non-competitive environment, which helps to improve diversity [7, 8]. To incorporate cooperative learning, the AEW program in CS100 follows the Engineering AEW philosophy [5] by paralleling the course syllabus, keeping labs at or above the level of the course. The AEW labs encourage diverse students to work together to solve challenging problems that follow the course material without needing to compete. The stronger students love the challenge, whereas the weaker students benefit from tutoring from their peers.

To date, the AEW in CS100 program has served primarily engineering-intended students by holding most labs through CS100M, the more "engineering" version of CS100. But, the computer science department and the engineering college wish to expand our capacity to hold more cooperative lab sessions for the broad range of students who might wish to learn about programming.

Unlike other AEW programs with labs, Cornell's CS100 does not have a designated/devoted computer laboratory due to lack of space and other resources. The current CS100 AEWs take place in the Green Room of Carpenter Library [9], which is a tremendous resource and terrific help that the engineering college provides. However, the space is not designed for cooperative learning, and so, facilitators and students have made frequent suggestions for a more suitable space.

We propose to design a computer lab that combines the aspects of cooperative learning with a computer laboratory course, a "cooperative learning computer laboratory." Modelled after labs with which other universities have begun to build [10], this lab would serve courses, such as CS100, that involve computer programming and software learning, all within the context of a cooperative learning environment.

Statement of Intent: Learning Outcomes

II. *What are the learning outcomes I am seeking? How will I evaluate the impact of the changes I have made? What are the indicators of improvement in learning and how will I measure them?*

Expanding the cooperative element in CS100 will hopefully do the following:

- reduce the fear from perceived competition by offering a "built-in" system of peer-based tutoring
- shift "external" competition to "internal" competition for each student, using the lab and other techniques
- encourage weaker students to persevere and practice, which is ultimately what helps a novice programmer
- channel exceptional students' energy and abilities into helping other students
- offer an opportunity for a broad range of students to work together
- help students build learning groups and partnerships that will benefit them in other courses, especially the first-year students
- demonstrate more exciting and invigorating examples to help motivate the course content
- increase the diversity of students for computer science and other technical fields
- train students for future undergraduate and graduate teaching positions as a result of the AEW facilitator training program

Ultimately, we hope that addressing the fear, competition, rumors, and otherwise "bad press," students will hopefully perceive an introductory computer-oriented course as broadening and less as a chore. An environment perceived as "friendlier" and less competitive might entice a more diverse range of students to take the course. In turn, a "happier" CS100 experience could broaden the students' opportunities for computer science and other technical fields. As computer skills become more integrated in other fields, labs in other technical courses will eventually adopt more required computer assignments. It would be worthwhile to see if the same aspects of computer science that turn away women and underrepresented minorities affect these other technical fields.

To evaluate the effectiveness of the cooperative element in CS100, we propose to measure the following:

- enrollment trends in the Fall and Spring semesters (students sometimes report that they worry more about the Fall because of the crop of CS-intended majors)
- enrollments in engineering and computer sciences courses that typically follow CS100
- enrollment in the CS100 AEWs
- exam and course grades for students who do and do not take the CS100 AEWs
- fear by taking evaluations at the beginning, middle, and end of the course
- drop out rates for AEW students and the class as a whole

The study should take place over at least four years so that we can compare numbers and demographics of incoming and graduating students.

Statement of Intent: Technology Applications

III. *What application of technology do I believe would help me to address the issues described above?*

The lab that CS100 currently uses, the Green Room [**9**], is a tremendous resource that the College of Engineering offers. But, unfortunately, the Green Room limits cooperative learning because it was designed to be a traditional computer lab. CS100 facilitators have noted the following problems:

- inadequate size, shape, and furniture: the students are somewhat cramped. Ideally, facilitators should be able to construct cooperative groups of two to four students, which means having the ability to place groups of students in eye contact in front of a computer. Moreover, the facilitators need to weave in and out of the groups. The long rectangular tables in the middle of the Green Room block students and facilitators from freely moving around.
- lack of privacy: besides having a glass partition with the connecting room, the entrance is open. Some cooperative learning exercises might be perceived as embarrassing from the participants, because of ice-breakers and other games. Spectating must be prevented.
- overabundance of computers: to foster group development, students actually need fewer computers. Given the Green Room's twelve computers, some students are frequently tempted into working alone and "tuning out" from the instruction. Fewer computers requires bigger groups, thus helping to maintain group integrity.

Based on our experience in the previous CS100 AEWs and limitations of the Green Room, the AEW program in CS100 has collected the following requirements for a new lab:

- space to hold an AEW session with sixteen students and two facilitators in which groups ranging from two to four students per group.
- flexibility of space to mix different groupings and arrangements of students, allowing students and facilitators to "roam around"
- furniture to support different group sizes and different activities, such as working in front of computers or "pencil-and-paper" development
- storage for textbooks and other materials for activities
- multiple forms of lecturing platforms

- privacy, because of group work (no exposure to the public)

To address these issues, the proposed lab would have the following characteristics:

- 625 s.f. of space, though about 550 s.f. would suffice with some modifications
- nine PC-workstations
- one projection unit and screen. Given enough space, there would also be two movable blackboards, and one fixed dry-erase board
- five desks/tables for holding the nine workstations, one table for a printer, and four movable tables for group work
- storage cabinets and other miscellaneous items.

The top view of a proposed cooperative computer lab is shown on page 8. Compared with similar labs [**10**], the lab places an emphasis on group activities, which often means that students need space to interact with each other and physically move around. Providing curved tables with and without computers should provide space for students in groups of up to four people to interact. Manufacturers, like **http://www.electronicclassroom.com/**, have developed such furniture to help create a cooperative environment. Other elements of the room, such as a projection system and printer, help to make the lab more flexible. The furniture will be flexible enough to offer alternative configurations that allow the same kind of cooperative interaction, as shown on page 9.

Given the flexibility of the space, the facilitators could either "wander around," where no group can effectively "hide." With multiple forms of lecturing available, the labs could teach in a variety of ways. Ideally the lab would also maintain a degree of privacy since cooperative labs sometimes involve "sillier" elements that the students might not wish to expose to the public

Because most students take classes in the south campus, the laboratory would be primarily used later in the day. Since its inception, the CS100 AEW staff has found that the students desire a two-hour block of time. So, we would schedule labs for CS100 and other courses in the afternoons from 12:30-2:30, and 2:30–4:30. Evening labs from 7:30–9:30 PM are also popular, with the exceptions of Tuesday and Thursday evenings when many prelims are held. We deliberately do not want to fill all times to provide other groups the opportunity to use the lab, as described below.

Lab Usage

The proposed cooperative computer lab will not serve just CS100. We envision multiple uses for the proposed lab, which include the following:

- An AEW lab for CS100: Given the enrollment of about 500 students per semester in two versions of CS100, we need space to coordinate several two-hour labs in the afternoon and evening.
- An AEW lab for CS211, which is the subsequent course to CS100 and will be piloting AEWs in Spring 2002: Given an enrollment of about 300 students per semester, there is potentially a large demand for future labs.
- CS99, which has a laboratory component built into the course. One future project involves the students broaching their fears of computing by building and programming robots with LEGO Mindstorms.
- Computational science AEW: The entire AEW program involves courses such as physics, math, and chemistry, which are many of the traditional introductory technical courses. Technical courses continue to increase the use of computer software for simulations, data acquisition, and experiments.
- Cooperative learning components in other courses: Currently, the College of Engineering coordinates the AEW program. Since many non-engineering students take these courses, we could provide an opportunity for "seeding" cooperative learning into other departments, such as writing workshops and software training. For instance, *ABEN 102: Introduction to Microcomputer Applications* might benefit.

- Student projects (Robocup, BRAIN, my computer game project, and others) that involve student teams that need to program and run/test software.
- Group tutoring sessions. Starting in Spring 2002, the College of Engineering is instituting weekly two-hour group tutoring sessions for the core engineering classes.
- Summer week-long CURIE [**11**] program for prospective women engineering students that includes computer-mediated learning activities.
- Occasional TA training sessions on teaching and technology.
- Occasional AEW facilitator training workshops.
- A research site for students conducting projects on technology use. Many students in the social and behavioral sciences conduct small projects on computer-mediated communication and resource use.
- Space-on-demand: By strategically placing a lab near the North Campus Residential Initiative, the first-year students will have a place nearby where they can work.

Implications

By constructing a cooperative learning computer lab, we have an opportunity to address many issues inside and outside Cornell.

Ultimately, a cooperative computer lab will assist with computer-oriented courses, especially the introductory programming courses. However, other courses with AEWs, such as physics, math, chemistry, and engineering, have tremendous software tools, which would be ideal for cooperative labs. Labs that show the practicality of the introductory courses help to boost enrollment.

Furthermore, technical courses need to address issues of extremely competitive environments. Women and under-represented minorities will drop computer science, in particular, because of perceived stereotypes of intense competition. So, by incorporating a lab built around cooperation, the residential areas will take an important step towards changing the impressions that decrease the diversity in enrollment in computer science and other technical fields.

Improving the environment also means that the lab should be made available for the students outside of class time, especially because of the proximity to the residential halls. Since the lab is designed to physically move the students into cooperative groups, groups of students would be encouraged to use the lab for their own studying and projects. Bringing undergraduate research teams into the lab, who also work intensely with software, will help to expose first-year students to the variety of research projects around campus.

The lab, itself, provides an opportunity to study the design and use of a cooperative lab. Though many studies site the strengths of cooperative learning, few explain how to infuse the same strategies into a computer laboratory environment. Studying how the various configurations affect cooperative learning and retention will provide outside exposure to Cornell, which, in turn, might show how Cornell is addressing the needs of diversity.

Besides attracting the current pool of women and under-represented minorities into math and science, Cornell could use the lab to increase the pool at the K-12 level. As part of outreach programs, we could invite K-12 students to Cornell to work and “play” in our lab. In turn, as we learn how to configure our lab, we can publish the details and help K-12 schools to develop similar environments, which help to improve diversity, overall.

Acknowledgments

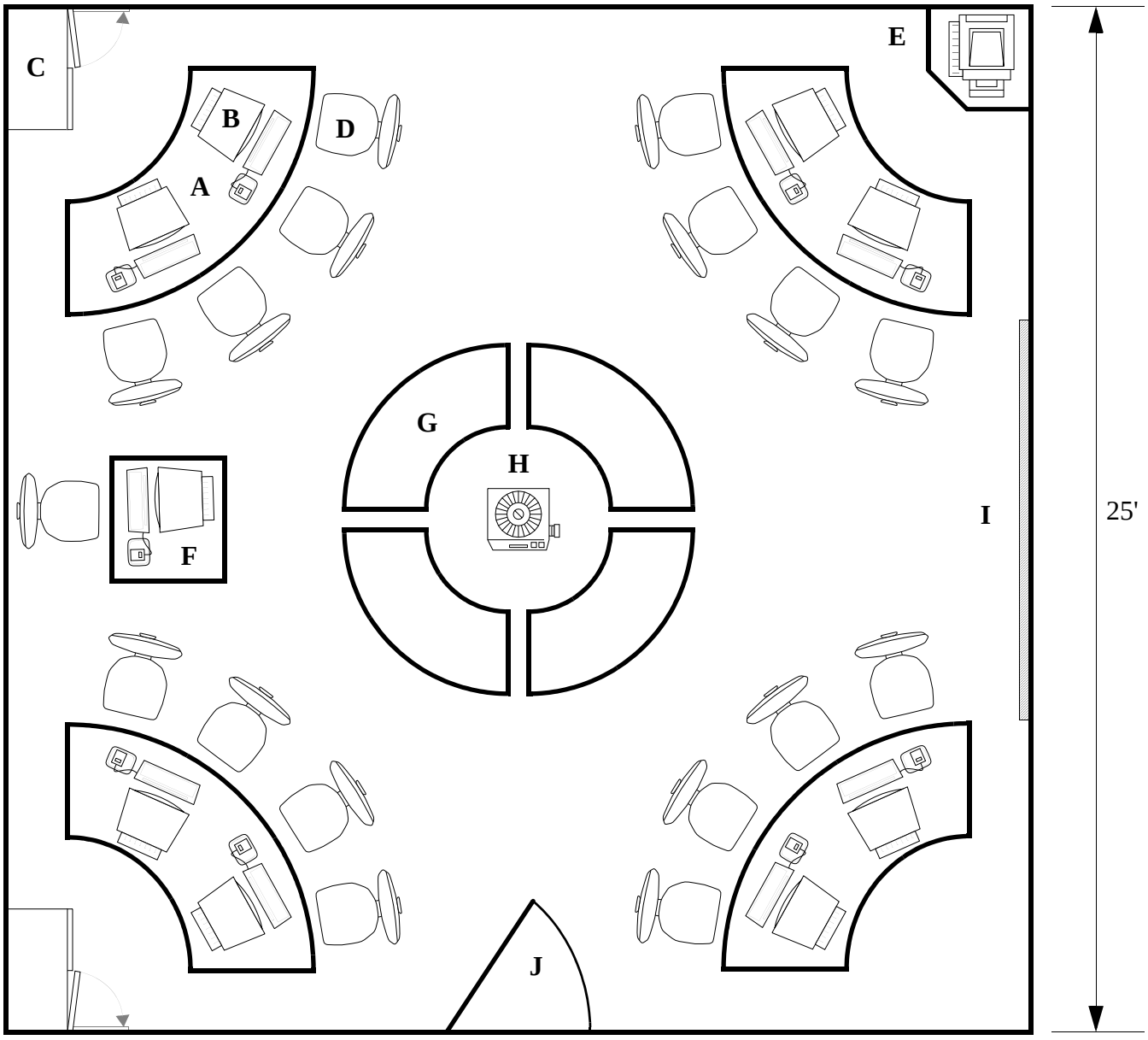
From September 15: I would like to thank Daisy Fan for her work as content liaison in from Fall 2000–Spring 2001 and contributions to the layout of the proposed CS100 AEW lab, Pat Spencer for her work with OR&IS, Sunny Donenfeld and Campus Life for encouraging us to create our lab, and CIT for their excellent classroom technology links.

For the updated portion: I would like to thank the entire team that has been working on this project since the previous version: Sunny Donenfeld, Daisy Fan, John Saylor, Tammy Shapiro, and Steven Weidner. Also, I would like to thank Jennifer Judson for the advice on the configurations on page 9 and the *Faculty Innovation in Teaching Grants Program* for helping us work towards making the cooperative lab a reality.

References

1. Camp, T., Miller, K. and V. Davies 2000. *The Incredible Shrinking Pipeline Unlikely to Reverse*, **http://www.mines.edu/fs_home/tcamp/new-study/new-study.html**, accessed July 23, 2001.
2. National Science Foundation 2000. *Women, Minorities, and Persons With Disabilities in Science and Engineering*, **http://www.nsf.gov/sbe/srs/nsf00327/**, accessed July 23, 2001.
3. Computer Science Department, Cornell University. *Choosing Your First CS Course*, **http://www.cs.cornell.edu/ugrad/FirstCourse.html**, accessed July 23, 2001.
4. College Board. *AP Computer Science – Introduction of Java in 2003-2004*, **http://www.collegeboard.org/ap/students/compsci/java_announcement.html**, accessed July 23, 2001.
5. Cornell Engineering College Office of Research and Instructional Support 2001. *Cooperative Learning through Peer Education*, **http://www.engineering.cornell.edu/studentServices/irs/aew.cfm**, accessed March 25, 2001.
6. Fan, D., Schwartz, D. and T. Yan 2001. *CS100M Academic Excellence Workshop*, **http://www.cs.cornell.edu/courses/cs100m/2001sp/aew.htm**, accessed July 23, 2001.
7. Johnson, R. T. and D. W. Johnson An Overview of Cooperative Learning (essay) **http://www.clcrc.com/pages/overviewpaper.html**, accessed July 24, 2001.
8. Trytten, D. A., 2001. *Progressing from Small Group Work to Cooperative Learning: A Case Study from Computer Science*, Journal of Engineering Education, January, 85-91.
9. Academic Computing Center Engineering Library. *ACCEL*, **http://www.accel.cornell.edu/**, accessed July 23, 2001.
10. Arizona State University College of Engineering and Applied Sciences. *Computer Equipped Cooperative Learning Classrooms*, **http://www.eas.asu.edu/~asufc/Classrooms/classroom.html**, accessed March 25, 2001.
11. The CURIE Academy, **http://www.engineering.cornell.edu/studentServices/womensProgs/index.html**, accessed January 18, 2002.

Proposed Cooperative Computer Lab



Scale:
1/4"= 1'

Legend:
A: computer table
B: workstation
C: cabinet
D: chair
E: printer
F: facilitator workstation
G: table
H: ceiling projector
I: whiteboard/screen
J: entrance

Alternative Computer Lab Configurations