1-1-2010

Lessons Learned From a PLTL-CS Program

Lessons Learned From a PLTL-CS Program

Abstract

The Peer-Led Team Learning (PLTL) approach has previously been shown to be effective in recruiting and retaining students, particularly under-represented students, in undergraduate introductory CS courses. In PLTL, small groups of students are led by an undergraduate peer and work together to solve problems related to CS. At Columbia University, the Columbia Emerging Scholars Program has used PLTL in an effort to increase enrollment in CS courses beyond the introductory level, and to increase the number of students who select Computer Science as their major, by demonstrating that CS is necessarily a collaborative activity that focuses more on problem solving and algorithmic thinking than on programming. Over the past five semesters, 68 students have completed the program, and preliminary results indicate that this program has had a positive effect on increasing participation in the major.

This paper discusses our experiences of building and expanding the Columbia Emerging Scholars program, and addresses such topics as recruiting, training, scheduling, student behavior, and evaluation. We expect that this paper will provide a valuable set of lessons learned to other educators who seek to launch or grow a PLTL program at their institution as well.

Keywords

Peer-Led Team Learning, Emerging Scholars Program, Out-reach, Women in Computer Science, Diversity, CS1

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-10-30.

Lessons Learned from a PLTL-CS Program

Christian Murphy, Rita Powell*
Dept. of Computer and Information Science
University of Pennsylvania
Philadelphia PA 19104
{cdmurphy, powell}@cis.upenn.edu

Kristen Parton, Adam Cannon
Dept. of Computer Science
Columbia University
New York NY 10027
{kristen, cannon}@cs.columbia.edu

ABSTRACT

The Peer-Led Team Learning (PLTL) approach has previously been shown to be effective in recruiting and retaining students, particularly under-represented students, in undergraduate introductory CS courses. In PLTL, small groups of students are led by an undergraduate peer and work together to solve problems related to CS. At Columbia University, the Columbia Emerging Scholars Program has used PLTL in an effort to increase enrollment in CS courses beyond the introductory level, and to increase the number of students who select Computer Science as their major, by demonstrating that CS is necessarily a collaborative activity that focuses more on problem solving and algorithmic thinking than on programming. Over the past five semesters, 68 students have completed the program, and preliminary results indicate that this program has had a positive effect on increasing participation in the major.

This paper discusses our experiences of building and expanding the Columbia Emerging Scholars program, and addresses such topics as recruiting, training, scheduling, student behavior, and evaluation. We expect that this paper will provide a valuable set of lessons learned to other educators who seek to launch or grow a PLTL program at their institution as well.

Categories and Subject Descriptors

K.3.2 [**Computers and Education**]: Computer and Information Science Education—*computer science education*

General Terms

Human Factors, Measurement

Keywords

Peer-Led Team Learning, Emerging Scholars Program, Outreach, Women in Computer Science, Diversity, CS1

*Dr. Powell is also an Extension Services Consultant for the National Center for Women & Information Technology.

SIGCSE'11 March 9–12, 2011, Dallas, Texas, USA

1. INTRODUCTION

Computer specialists are among the occupations projected to grow the fastest and add the most new jobs through 2018, with job prospects in this industry the greatest for those who hold the bachelor degree. The U.S. Department of Labor estimates that there will be 1.4 million technology jobs in 2018, but U.S. universities will be able to provide graduates that qualify for less than 1/3 of these positions.[1] In addition, diversity in computer science is decreasing with 18 percent of bachelor's degrees in computer science awarded to women in 2006, and fewer than 12 percent at U.S. research universities.[2] This trend underscores the scarcity of diverse insights in creating new technologies, and calls for an intervention through pedagogy and curriculum to attract a larger sector of college students to computing majors.

One effort to recruit and retain students, particularly under-represented students, in undergraduate introductory CS courses is the Emerging Scholars Program (ESP), based on Peer-Led Team Learning (PLTL). Since many students are not exposed to computer science in high school, introductory CS courses (hereafter CS1) are often their first experience. Large, lecture-based classes and complex one-person programming assignments can be daunting to students with no CS background. However, ESP/PLTL encourages active learning, as opposed to the common paradigm of passively listening to lectures in an auditorium filled with hundreds of students. ESP/PLTL supplements traditional science, technology, engineering, and mathematics (STEM) courses in that students meet in small groups (6-8 people) that are facilitated by a student who has recently completed the course. The peer leader presents problems to the group, and then guides students as they collectively brainstorm, discuss and analyze the problems to come to a solution.

Since 2008, Columbia University[3] has offered an ESP/PLTL program for students in its introductory CS course. This enrichment program, called the Columbia Emerging Scholars Program (CESP), is not directly tied to CS1 course content, but is rather aimed at demonstrating to students the breadth of CS topics, using fun and interesting group problem-solving activities. Initial results indicate that CESP has been successful in its goals of increasing retention of students (i.e., the number of students who continue on to CS2) and increasing the number of students who major in CS.

This paper describes our experiences (both positive and

[1] http://www.bls.gov/news.release/ecopro.t06.htm
[2] http://www.cra.org/resources/taulbee
[3] The first author of this paper was previously a graduate student at Columbia.

negative) in launching, expanding, and refining CESP over the past three years. Whereas other authors have discussed the theoretical and observed merits of such a program [2, 6, 8], or focused on how to get a program off the ground [4], our intent is to share the general lessons learned so that others who are starting, growing, or modifying a PLTL program can learn from our experiences.

2. BACKGROUND

The ESP concept was initially developed by Uri Treisman at the University of California, Berkeley, in the 1970s to propel the mathematics skills of African-American students by providing them with an honors workshop adjunct to the introductory calculus course to solve challenging problems in group study sessions [7]. Originally called the Math Workshop Program at Berkeley, the program became the Emerging Scholars Program at the University of Texas, Austin, and has been adopted by over a hundred universities and expanded to support courses in the natural and physical sciences and engineering.

ESP/PLTL has long been applied successfully in other STEM disciplines - for instance, the PLTL Workshop Project at City College of New York has been using it in chemistry since the early 1990's. A typical ESP/PLTL program consists of a set of weekly one- or two-hour long workshops adjunct to an academic course, led by an undergraduate workshop leader, or "peer leader", who has been specially trained in PLTL techniques. Workshops may have anywhere from as few as six to as many as 15 participants, and the students work together to solve problems, without the pressure of exams or grades. Over the course of the semester, weekly meetings are held between the peer leader and program coordinators (who may be graduate students, faculty members, or administration) to ensure that the program is on track, that educational goals are being met, and that the peer leader is sufficiently comfortable with the workshop material. At the end of the semester, the coordinators typically perform some evaluation procedure to measure the success of the program, and to make adjustments as necessary.

In Computer Science, ESP/PLTL has been growing in popularity over the last few years due to the success of the Emerging Scholars Program in CS (ESP-CS) endeavor. Funded by an NSF grant, eight universities worked closely together to develop, implement and evaluate a PLTL program for CS1 over three years. They have published detailed findings of their experiences and results [3], as well as modules of course content, to assist other CS departments in starting similar PLTL programs.

In addition to using PLTL, ESP-CS uses targeted recruiting to attract students from under-represented groups. Students who may otherwise be turned off by the stereotype of computer scientists as lone programmers can discover through PLTL that CS is a collaborative discipline that encompasses much more than just coding. Initial studies have shown that students, particularly students from under-represented groups, are much more likely to be drawn to STEM subjects when the material focuses on teamwork rather than on working alone [1, 5].

3. ESP/PLTL AT COLUMBIA

At Columbia, the CS1 course is a large, lecture-style class with a single section of 150-200 students. There are no lab or recitation sections, so students do not have the opportunity to meet in small, structured groups. Many CS1 students have not yet declared a major, since Columbia students typically do not declare until their second year, and the course consists of about 40% women - a much higher proportion than the CS department as a whole, where in 2008 only 16% of majors were women. Clearly many of the female students who complete CS1 are going on to choose other majors. Our objective, then, was to increase female participation in CS beyond the introductory level, by addressing some of the issues that other researchers have claimed may be driving those students away.

On the heels of the successful "PLTL in CS Workshop" at Duke University in 2007, the CS Department at Columbia University funded a pilot program in Spring 2008 to encourage more women to pursue CS as a major. By exposing students to a variety of CS topics early in their education, focusing on collaborative problem solving instead of technical details or coding, and creating a network of undergraduate women interested in computing and technology, the Columbia Emerging Scholars Program aims to increase the number of female students who go on to take CS2 and, ideally, choose CS as their major. CESP is an enrichment program, not directly tied to CS1 course topics, in which students meet for one hour per week in small groups to solve problems, facilitated by undergraduate peer leaders. There are nine workshops over the course of the semester, followed by a reunion event with former CESP participants, which provides a networking opportunity for students, recruiting opportunity for future CESP leaders, and a way to advertise other groups and events in the department.

During the pilot program, two graduate students (the first and third authors of this paper), one faculty member (the fourth author of this paper) and one undergraduate peer leader worked closely to come up with engaging problems from a variety of CS disciplines, including natural language processing, information retrieval, and social networking. The pilot was very successful. In the pilot program's exit survey, all six students said they would recommend ESP to a friend. One student wrote, "These workshops gave me a better perspective of what computer science is. I have learned that it is extremely useful and pertains to problems and issues that are in our daily lives."

Thanks to a seed grant from the National Center for Women & Information Technology (NCWIT) and Microsoft Research, CESP grew to two sections in Fall 2008, and we have continued to hone and expand the workshop content over the past two years. In total, 68 undergraduates have participated in CESP as students, and 10 have gone on to be peer leaders and/or workshop assistants.

The Columbia Emerging Scholars Program differs from similar programs at other institutions in a variety of small but significant aspects:

- Rather than being open to the general population of CS1 students, CESP is limited to 12-15 students per semester, meaning that students must go through an application process in order to participate (Section 3.1).
- In its first three semesters, CESP only allowed for female participants; this has since changed, though, and male students are now invited to participate as well (Section 3.2), and a gender balance is maintained in the coed workshops.

- CESP workshops feature no programming-related material at all, and focus only on CS as an algorithmic problem-solving activity (Section 3.3).

The rest of this section describes these differences further, discusses their influence on the success of the program, and provides suggestions for other educators who are considering developing similar programs.

3.1 Selection

CS1 at Columbia University is a requirement for many engineering students, and acts as a service course for many other departments, so it is taken by students from a variety of backgrounds, including upper-class students who have already declared majors. On the other hand, CESP is explicitly aimed at students who are undecided or potentially interested in CS as a major, so CESP is only appropriate for a subset of CS1 students.

Students apply to CESP via an online form, in which they are asked to indicate the level of their CS background, the majors that they are considering, and how likely they are to take any further CS courses. They are also asked open-ended questions like "why do you want to participate in this program?"

The target CESP candidate is undecided as to her major, and curious about CS. The ideal CESP section has students from diverse backgrounds (some with prior CS experience, some new to CS), with most students from the freshman and sophomore years, if possible.

Typically, we would try to avoid selecting: students who have already decided to major in something else, since this program is not targeted to them; students who seem *too* enthusiastic about CS, since we do not want them to overwhelm or dominate the workshop discussions; and continuing education, upperclassmen, or graduate students, who may be older (sometimes significantly so) than the other students and even the workshop leader.

There have, of course, been incidents when students were not selected for CESP and were very disappointed. It is important that the selection criteria is clearly defined, in case a student escalates her complaint about not being chosen. It is not necessary to publicly declare the selection criteria, but it should be something that is defensible, just in case.

At other institutions, ESP is open-enrollment, i.e., available to all students in the CS1 course who wish to participate; in some, ESP is a mandatory corequisite to CS1. Although these approaches broaden the number of students who are exposed to the benefits of ESP, they may not be appropriate in all cases. At Columbia, the CS1 course typically has up to 200 students per semester; certainly not all of them will benefit from CESP, either because they do not have the right personality to enjoy its collaborative problem-solving nature, or they do not have interest in CS beyond learning how to program. In our case, then, it is preferable that students should have the option of being a part of CESP, which we feel will reduce the number of participants who are merely "along for the ride," and improve the quality of the workshops.

3.2 Coed vs. Women-Only

The initial incarnation of CESP was specifically designed for female students. We feared that male students might tend to dominate the workshops, and wanted to provide the female students with a more comfortable and less confrontational environment. Additionally, we sought to create a community of young women in the CS Department, and use CESP as a stepping stone to other organizations, such as Women in Computer Science and ACM-W.

Restricting CESP to only female students does, of course, raise a number of issues. First, what are the educational, ethical, and legal implications of creating an enrichment program that is limited only to women? We struggled with this issue, fearful that male students would claim that they were not being given equal opportunities. Fortunately (if not somewhat surprisingly), this situation never arose. In fact, university administration supported the program financially under the "diversity" umbrella. However, whether or not a female-only program is acceptable/allowed may differ at other institutions.

Another concern is that the program would be seen - particularly by participants - in a derogatory manner as "remedial Computer Science" or "Computer Science for Girls". It was important for us to address this immediately in the advertising and recruiting efforts, as students may shy away from the program if they feel there is some stigma attached to it. We pointed out that CESP workshops are *not* extra office hours or tutoring sessions, but rather that they are problem-solving sessions with topics that are related to CS, though not directly related to the CS1 course material. We found that by focusing on the enrichment aspect of the program, and the advantages that it provides to students outside the classroom, students quickly realized that they were being presented with a wonderful opportunity.

As we sought to grow CESP, we considered opening the program to male students as well. Before doing so, we polled the students who had participated in the female-only semesters of CESP, and asked "How much did you like having all-female workshops?" We were surprised to find that the number of students who wanted female-only workshops (46%) was about the same as the number who would have been fine with coed workshops (42%). Thus, starting in Fall 2009, we opened up the program to male students, with one all-female section and one coed section, and the program is now fully coed.

3.3 Topics

Since CESP is only open to a subset of students in the CS1 course, we felt it was inappropriate to have the workshops be tied directly to course material, as it would give the participating students an unfair advantage over those who were not in the program. Instead, we try to demonstrate the breadth and variety of fields within CS. The ESP workshop topics we developed include: designing algorithms; encoding and encryption; machine translation; human-computer interaction and usability; social network analysis and graph theory; and hard (NP-complete) problems. These workshops are designed to engage the students in group problem solving, so that they can see that CS is a collaborative activity that is focused on problem solving, not solely on programming.

Many programs similar to CESP include topics related to the programming languages they are learning in their CS1 courses, such as exploring the language's syntax and semantics, how the language works under the covers, or how to implement an algorithm using the language and its APIs. However, we chose to focus workshop content solely on algorithmic thinking and problem solving, since students

do a lot of programming in the course already. Although the CS1 course at Columbia is taught using Java, CESP does not include any material related to how to use Java or how Java works. All of the CESP workshops are language-independent.

The students seem to agree with this notion: 50% said they would prefer not to have Java programming covered in the workshops, compared to 33% who said they would. One student preferred not to program in CESP because it taught her that "much is done before programming begins," such as "simply thinking about how to solve the problems." Another agreed that the "thinking aspect [of CS] can get forgotten, lost, trampled over or almost done away with when programming." Last, a student pointed out that she "liked how we focused on theory by just thinking of solutions rather than programming them. The programming language seems more like a chore and doesn't interest me, whereas the theory behind the actual computing does."

Last, we also considered giving the students take-home assignments, either to do in preparation for an upcoming workshop, or to further elaborate upon material covered in previous ones, as is done in ESP programs at some other institutions. Regardless of whether these assignments would be graded or not, or if they were to be done alone or in groups, fully 85% of the students indicated that they would not want take-home assignments included in the program. However, we often provided students with handouts at the end of each workshop, in which we listed books and websites that students could look at if they were interested in learning more about the day's topic.

4. LESSONS LEARNED

As it is the goal of this paper to share our experiences with educators looking to start, grow, or modify an ESP-CS program, this section highlights other lessons we have learned through the administration of CESP.

4.1 Recruiting

Recruiting of students is essential to the program's success, as it is important to attract students who are suited to its open atmosphere and workshop format, but are also not quite sure that they will proceed on to CS2.

One lesson that we have learned is that what appears to be an effective recruiting technique in one semester may not be so effective in another. For example, in the CESP pilot, we announced the program in class and sent out a recruiting email to the female students. We had 19 applicants for six spots, and the students who participated were extremely enthusiastic and since then have been actively involved in programs for women in CS. A year later, though, using the same recruiting techniques, only 15 students applied for 12 spots. As a result, three out of the 12 accepted students had already declared majors other than CS.

At times we have been disappointed with the lack of applicants who are truly interested in CS. In the past, we have sometimes accepted these students into the program to fill out the numbers, but this can backfire. Students who are not particularly interested may refuse to participate, sleep during workshops, or act out, which lowers the morale of the entire workshop. It may be necessary to reduce the number of participating students and then either have smaller workshops (which seems to be preferable) or have one workshop but with a larger number of students.

We have found that in-class announcements can be an effective recruiting technique, and work best when undergraduates (i.e., current workshop leaders and/or last semester's workshop leaders) give part of the talk. One of the best strategies was getting the RA in one of the science/engineering dorms to tell all the CS1 students on her floor. That is, students are more likely to be interested in such a program when they hear about it from a peer, rather than from a graduate student or faculty member.

We suspect also that the program would attract more students if it were institutionalized as a proper course that appeared in the registrar's bulletin of classes. At Columbia, CESP is an extracurricular program for which participating students get one unit of "research credit"; this allows us to select the students we find desirable and to limit enrollment. However, in some cases, freshman advisors who were not familiar with the program have expressed concern about their students participating in CESP, since it did not appear to be a "real" course. Additionally, although the students received one unit of credit, they could not apply it towards the graduation requirements for being a CS major in some cases. If CESP had its own course number and had a description in the course catalog, these problems may be overcome, and more students may choose to apply and participate.

4.2 Setting Expectations

In the early days of CESP, we struggled to properly convey during recruitment what, exactly, CESP was. Whether we advertised it in person or in an email, we knew we only had a few short moments to make a positive impression on the students, who may have thought to themselves "that sounds uninteresting to me" or "the last thing I need is yet another event in my schedule". Even though we specifically focused on the facts that CESP workshops consisted of small groups, that there would be only problem solving and no programming, and that CESP would be a lot of fun, it is possible that some students were turned off by notions that CESP would be tantamount to extra office hours or, worse, extra homework assignments.

Even the students who *did* decide to participate may have had incorrect notions of what CESP entailed. Wrote one in her exit survey, "Based on the descriptions, I had thought it would be more of a lecture style rather than small workshops. I like the way it turned out, but I had no idea when I applied." Wrote another, "I expected there to be more ... programming. It was more that we tackled basic puzzles, [and] just talked about solutions as opposed to what would actually go into a computer [program]."

As mentioned above, we have found that students tend to be more interested in CESP if they hear about it from a former participant, rather than from a graduate student or instructor. This may help in setting the expectations, too: even though the message is the same, it may sink in more when it comes from a peer. Since we started exclusively having undergraduates advertise the program, almost all students have said that CESP matched their expectations, compared to much lower numbers in previous semesters.

It has also been suggested that an "open house workshop" be held for all interested applicants, so that they can see for themselves what a CESP workshop is like before deciding whether to apply. We have not yet tried this but suspect it would at least give the students a solid idea of the PLTL concept, and may help increase the number of applicants.

This may require some juggling of schedules, since ideally all participants would apply to, be accepted to, and sign up for CESP before the "add class deadline" (which is at the end of the second week of the semester at Columbia). It may be necessary to do this very early in the semester, perhaps even before first CS1 class meeting.

4.3 Scheduling

In the pilot program in Spring 2008, when we only had six participants, we first selected the students and then tried to find a time for workshops that fit into everyone's schedule. This, of course, does not scale particularly well. In the second semester of CESP, we accepted the top 15 applicants, regardless of their availability, and tried to balance them across two meeting times; however, we ended up with 11 students in one section and just four in another, which clearly was not desirable.

At our university, and supposedly at most others, Friday afternoon workshops seem to work for just about everyone. So now, just like a regular class, we select two times on Friday (maybe early afternoon and late afternoon) and then tell applicants "these are the times". We choose workshop leaders and assistants who can make those times and then, in the application for participants, students choose which time(s) they want to apply for. If they can't make it to either, then they don't get in. This runs the risk of excluding highly-desirable students just because of their availability, but is the only feasible approach for programs of any size greater than just a handful of students.

It is important to work out the weekly meeting times and the semester-long workshop schedule in the beginning of the semester. In CESP, there are nine workshops plus the end-of-semester reunion and party, so there is some wiggle room in the schedule. We have had to be particularly aware of religious holidays (many of which may be observed by the students but not by the university), the Grace Hopper conference, the course midterm, and exam review sessions. By scheduling around these events, we try to minimize students' absences and ensure a consistent attendance rate, which improves the flow of the workshops.

Scheduling is an issue that concerns not just the participating students, but the workshop leaders as well. When selecting peer leaders, we seek students who have enough time at reasonable hours to make both the workshop and any weekly coordinators meeting. As pointed out previously, the best approach has been to set the times in advance, and then find a student whose schedule allows her to be free at those times. In the early days of CESP, we worried that would would not be able to find qualified peer leaders, and in one semester selected a student who had a number of other commitments and did not have a very flexible schedule. Despite the student's enthusiasm and winning personality, she simply was not able to attend all meetings and could not spend adequate time preparing for her workshops. This is definitely a situation to avoid.

4.4 Training

Ideally, the peer leader would be a student who had previously participated in CESP, so that she is already familiar with the PLTL style as well as the material that is covered in each workshop. Of course, familiarity does not necessarily equate to competence or expertise, so it is essential that adequate time is invested at the beginning of the semester to training peer leaders and having them lead sample workshops. This is typically uncomfortable and awkward for the student, but is the only way to be sure that she is ready.

This training should continue over the course of the semester. During weekly coordinator meetings, we try to make sure that peer leaders are comfortable presenting the content themselves. Of course, if one asks a student "are you okay with this material?", she will almost always answer "yes", just to avoid having to present it in front of graduate students or faculty members, even if she really is not prepared. So we have the leaders present the material at the weekly meeting as if they were running the real workshop. This is the only way to know that they really are ready.

4.5 Behavior and Attendance

At the beginning of CESP, students are given a contract that lays out the expectations for how they will behave and participate in the workshops. This contract addresses issues related to using laptops and cellphones, and this policy is emphasized from the very beginning. In CESP, the students have one task and one task only: show up and pay attention for 60 minutes. That's it. No homework, no exams, no preparatory work. So it should be possible for them to be attentive for one hour a week without sending text messages or checking Facebook. There is no reason for students to be using cellphones or laptops during the workshops. A student who is not participating because she is distracted thus becomes a distraction to all other students, as well.

The CESP contract also stipulates that any lateness of more than 10 minutes is considered an absence, and more than two unexcused absences will result in a failing grade (CESP participants earn one point of pass/fail credit). In the very beginning, we emphasize to all students that it is important that they are on time for the workshops, as they cannot really get started until all participants are present.

There have been some occasions when we had to give students a failing grade because they missed (or were significantly late for) more than two workshops. We notify students after the second absence that they are in danger of failing if they miss another workshop or are very late. This adds a bit of administrative overhead during the semester, but will probably help avoid a difficult and uncomfortable situation down the road.

4.6 Discussion vs. Problem-Solving

As discussed above, CESP workshops focus on the algorithmic thinking aspects of computer science, in which students need to work together to devise a solution to a particular puzzle or problem. Often there is a "correct" solution to these problems, though in many cases there may be more than one solution or approach, and part of the workshop is to discuss the tradeoffs between the approaches.

We recently introduced a more open-ended, discussion-based workshop in which the students were asked to consider social issues related to computer science and technology, and then come up with legal or academic policies that made sense to them. We expected that the students would like this workshop very much, as it included topics that are familiar to the students, such as digital copyright laws and online plagiarism. However, in the exit surveys, the workshop was rated second-lowest (out of nine workshops) by the students. Surprisingly, some of the students commented that the workshop did not appear to be about computer sci-

ence. This was very unexpected, and indicates that we need to broaden the students' concept of what "computer science" is and the issues that they, as computer scientists, will need to address in the future.

4.7 Evaluating the Program

The only way to know whether an Emerging Scholars Program is having a positive impact is to have a formalized way of tracking the students and accumulating both qualitative and quantitative data about their experience in ESP, CS1, and beyond.

At Columbia, all CESP participants fill out online surveys at both the beginning and end of the program that ask questions such as "how likely are you to take another CS course?" or "how likely are you to major in CS?" Ideally, the participating students' likelihood of taking another CS course or majoring in CS would rise after completing CESP, of course. And, supposedly, the increase in likelihood would be greater than the increase (if any) for those students *not* participating in the program.

Note that students' self-reported likelihood of taking another CS course or majoring in CS does not necessarily equate to *actually* taking another CS course or majoring in CS. However, gathering data about students' activities after they leave CS1 often requires assistance from the department or school's administration, and may bring about issues related to privacy. In order to collect that sort of data, we have had to get administrative buy-in beforehand, and make it clear that we wish to be able to answer questions such as "what percentage of CS1 students who did not participate in ESP go on to take CS2?".

For collecting other qualitative data, such as "how much of an impact did ESP have on your decision to take CS2 (or major in CS)?", it is important to stay in touch with ESP alumni so that they will be willing to provide such feedback. In our case, we sent out a survey to all former CESP students (some of whom had participated in the program two years prior), asking questions about the CS courses they had since taken and their impressions of CESP in hindsight, but had only a 33% response rate. Perhaps we would have had a higher response rate if we had set the expectation that we would later on ask such questions, rather than sending out the survey after a long period of no communication at all.

5. RESULTS

Fall 2010 marks the sixth semester of CESP offered as an adjunct to CS1 at Columbia University. Initial evaluation of CESP's impact on enrollment indicates that CESP is increasing women's participation in the computer science major at Columbia. Women's percentage of total computer science majors increased from 9 percent in 2007 to 21 percent in 2010, and to date 45% of CESP students who have declared a major chose Computer Science.

We have also observed that peer leaders benefit greatly from the program, as well, as they grow and gain confidence in themselves as computer scientists. To date, two former CESP peer leaders have participated in CRA-W's Distributed Research Experience for Undergraduates program, and one has earned an honorable mention for the CRA Outstanding Undergraduate Researcher award. Another currently is a software engineer at Microsoft. By all these measures, we consider CESP to be a success.

6. CONCLUSION

Emerging Scholars Programs, based on the Peer-Led Team Learning methodology, have been very effective in improving retention in CS courses and increasing the number of students who choose CS as their major. However, starting, growing, and maintaining a successful program can be a tricky task. In this paper, we have described the Columbia Emerging Scholars Program, from its inception through its expansion, and discussed some of the lessons we have learned along the way. From recruiting and selecting students, to training peer leaders and scheduling meetings, to deciding on workshop topics and evaluation criteria, our experiences have been largely positive and the program has been a success. We hope that this paper will help other educators looking to start programs of their own.

7. ACKNOWLEDGMENTS

The authors would like to thank Remi Moss for her assistance with data collection, as well as Henning Schulzrinne, Shree Nayar, Julia Hirschberg, and Kathleen McKeown for their support and encouragement. We especially extend a special thanks to all the students who have participated in CESP. The Columbia Emerging Scholars Program is funded in part by the National Center for Women & Information Technology and Microsoft Research.

8. REFERENCES

[1] J. M. Cohoon. Toward improving female retention in the computer science major. *Communications of the ACM*, 44(5):108–114, May 2001.

[2] L. Gafney and P. Varma-Nelson. *Peer-Led Team Learning: evaluation, dissemination, and institutionalization*. Springer, 2008.

[3] S. Horwitz and S. H. Rodger. Using peer-led team learning to increase participation and success of under-represented groups in introductory computer science. In *Proc. of the 40th ACM technical symposium on computer science education*, pages 163–167, 2009.

[4] S. Huss-Lederman, D. Chinn, and J. Skrentny. Serious fun: peer-led team learning in CS. In *Proc. of the 39th SIGCSE technical symposium on Computer science education*, pages 330–331, 2008.

[5] R. M. Powell. Improving the persistence of first-year undergraduate women in computer science. In *Proc. of the 39th SIGCSE technical symposium on Computer science education*, pages 518–522, 2008.

[6] C. Stewart-Gardiner. Using peer led team learning to assist in retention in computer science classes. *Journal of Computing Sciences in Colleges*, 25(3):164–171, January 2010.

[7] U. Treisman. Studying students studying calculus: A look at the lives of minority mathematics students in college. *The College Mathematics Journal*, 23(5):362–372, Nov. 1992.

[8] T. T. Utschig and M. Sweat. Implementing peer led team learning in first-year programming courses. In *Proc. of the 38th Annual Frontiers in Education Conference*, pages F3C–13–F3C–18, 2008.