LEVERAGING EXISTING OUTREACH PROGRAMS TO REACH UNDERREPRESENTED MINORITIES

Maribel Gonzalez, Lucy Vasserman, Sara Owsley Sood, and Tzu-Yi Chen
Computer Science Department
Pomona College
Claremont, CA 91711
909-607-8651
{maribel.s.gonzalez, lucy.vasserman}@gmail.com
{sara, tzuyi}@cs.pomona.edu

ABSTRACT

Because majoring in computer science often requires taking a CS1 course by the end of one's first year in college, high school students need to be made aware of computer science as a potential major so that they can plan their college coursework accordingly. We describe two robot-based activities that were offered as part of a broader summer program aimed at preparing local high school students from groups traditionally underrepresented in higher education for college. The goal was to give these students a more accurate understanding of what computer science is and what computer scientists do, to share our excitement about computer science, and to get them to think about taking a CS1 course their first year in college. While both activities were good for exposing students to computer science, students' experiences in the two were quite different.

INTRODUCTION

The importance of increasing the participation of underrepresented minorities in computer science specifically, and science and technology more generally, has been documented in reports such as [4, 8, 9]. Unfortunately, the data shows that STEM fields are losing underrepresented minorities at every stage of the pipeline. Solutions remain elusive, but one approach is to target transition points: encouraging high school students to go to college, encouraging undergraduates to go to graduate school, and so on.

Computer science faces a particular challenge in that, despite the efforts of groups such as the CSTA (http://csta.acm.org), computer science is poorly represented in most high school curricula. Because students go to college without exposure to computer science, they may not consider it as a potential major, and therefore may not take a first computer science course early enough in their college career to complete a computer science major. Ideally all high schools will eventually require a course in computer science. In the meantime, efforts have been directed towards extra-curricular activities and weekend or summer programs that would introduce students to various aspects of the computing field [6, 7]. However, these programs typically solicit applications and focus primarily on computer science. As a result, they may appeal only to students who are already interested in computer science, perhaps through some even earlier experience. A broader approach is to incorporate computer science content into outreach programs that are designed for students interested in math and science [5]. This has the advantage of introducing computer science to students already considering math, science, or engineering degrees. However, it may do little to increase overall interest in STEM fields. Yet another approach, and the one we explore here, is to reach even more broadly by working with outreach and bridge programs that are meant to help prepare students for college.

In this paper we describe our experiences working with an established summer program at Pomona College that prepares students from groups traditionally underrepresented in higher education to be admitted to, and to succeed in, college. We believe the strategy of partnering with an existing outreach program is promising as a way of encouraging a population that is already prepared to succeed in college to explore computer science once they get there. In the rest of this paper we describe the design of two robot-based activities, discuss results gleaned from pre- and post- surveys given to the participants, and finally comment on lessons learned.

BACKGROUND

The Pomona Academy for Youth Success (PAYS) is a 4-week rigorous summer program which serves up to 90 students each year. These students are rising sophomores, juniors, and seniors at local high schools and are from “groups traditionally underrepresented in higher education—students who are first in their family to attend college, those from low income families and those who are African American or Latina/o.” [1] The students take critical thinking and math classes taught by local faculty, enroll in short elective courses taught by college students (many of whom are alumni of the program), and receive guidance on navigating the college admissions process. Students are encouraged to take part in the program for more than one summer; indeed, all the rising seniors are participating for either their second or third summer.

In addition to the above activities, rising seniors take part in an intense short-term research mini-project. These projects are supervised by faculty across the college and have included projects in disciplines such as theater, sociology, chemistry, and mathematics. Rising juniors and sophomores can sign up for optional activities in a variety of areas that have, in the past, included preparing a newsletter or working in an organic garden. The PAYS staff assigns students to the activities and projects available each summer after taking into consideration the students' rankings of the different offerings and the instructors' requests about features such as class size. Because the activities are optional, students can choose to not participate in an activity or request to switch activities. The assignment to research mini-projects, on the other hand, is final.

STRUCTURE OF THE COURSES

In the summer of 2008 one of the authors led a group of 5 rising seniors on a research mini-project related to finding good fill-reducing orderings for sparse matrix computations. Students were taught basic commands in Matlab, and then asked to experiment with scripts that compared the behavior of different orderings. In hindsight, that task was too far removed from anything the students had experience with, and so they had difficulty seeing the big picture or understanding why the work was important. Indeed, the students showed significant enthusiasm and interest when the professor gave a mini-lecture on applications of sparse matrices [10].

As a result, for the summer of 2009 we made the topic more hands on by having the students learn Python in order to work with Scribbler robots (http://www.parallax.com/tabid/455/Default.aspx). In addition, we offered not just a mini-project for rising seniors, but also a robot-based activity for rising sophomores and juniors.

Activity Group

This group of 15 rising sophomores and juniors met with us twice a week for an hour each time. Each session began with a brief presentation on robotics or more generally AI. The students then worked in groups of three on learning basic Python programming with the goal of being able to issue simple commands to Scribbler robots. Projects included getting the Scribbler robots to draw simple geometric shapes.

Research Group

This group of 6 rising seniors met with us four times a week for 2 hours each time. Because we wanted them to help each other and to start to view computer science as a collaborative field, we had them work in pairs. The students chose their partners and kept the same partners over the four weeks.

The curriculum closely followed the material in [2]. The first week and a half was highly structured: in each session an undergraduate instructor gave a mini-lecture and the students then worked in pairs on worksheets with exercises such as those in [2]. The students also periodically worked as one group of six to do activities such as getting the Scribblers to write out the word "PAYS" and having the Scribblers race around the room with each robot controlled by one person who could see the robot and another who could only enter commands.

Halfway through the second week we asked each pair to choose a final project. We gave them a list of possibilities, together with our thoughts on the likely difficulty of, and challenges presented by, each. Ultimately the three pairs chose the following projects:

- burglar alarm: programming the Scribbler to emit a beeping sound when it detected a door opening
- light seeker: programming the Scribbler to detect and follow a light source
- line follower: programming the Scribbler to follow a line on the ground

The line following project was the most difficult, and that pair never got their program to fully work. The burglar alarm was the most straightforward, so that pair ended up spending time adding extra features such as the ability to play songs in place of simply beeping. The light seeker program was somewhere between the other two in difficulty, and that pair had no trouble finishing. In fact, they spent time trying to adapt their program to detect and head towards objects of a specific color.

In the last week we switched to preparing presentations. The students first gave a 12-minute Powerpoint presentation to the rest of the PAYS community (a group of over 100 high school students, professors, and staff). The final presentation had six content slides, with each pair of students responsible for presenting a slide on their final project and a slide on some aspect of the overall experience. We used the slides from the talk to prepare a poster that was displayed at the closing ceremonies.

EVALUATION

We administered pre- and post-surveys to all the participants. The surveys contained a dozen Likert items meant to evaluate students' attitudes, competencies, and confidence. Examples of items include:

- Computer Science is useful

- Computer Science is fun
- I am confident in my problem solving ability
- Computer Science is something I could be good at

In addition, there were several free-response items including:

- What do you think Computer Science is?
- What do you think Computer Scientists do?

The six research students completed both the pre- and the post-surveys. All 15 activity students completed the pre-survey, but only 9 of them completed the post-survey.

We emailed the research students during their first year of college to find out whether they had taken, or were planning on taking, a CS1 class.

RESULTS

The following table shows the average score on 7 of the 12 Likert items for both the pre- and post-surveys, for the students in the research mini-project and for the students in the activity. A 1.0 corresponds to "strongly disagree" and a 5.0 to "strongly agree".

	research		**activity**	
	pre	**post**	**pre**	**post**
Computer science is fun	3.0	4.8	3.9	4.2
Computer science is challenging	3.8	4.2	4.1	5.0
I would consider majoring in Computer Science	2.2	3.0	2.8	3.2
Computer Science is hard	4.0	3.3	3.5	4.1
Computer Science is useful	5.0	4.5	4.6	5.0
I am confident in my problem solving ability	4.0	3.8	3.9	3.8
Computer Science is something I could be good at	3.5	3.7	3.5	3.7
I am a capable programmer	2.0	3.2	3.2	3.6

Overall we saw little change between the pre- and post-surveys on items meant to evaluate students' self-confidence. We saw some differences in other items between the students who participated in the research project and the activity. For example, the average score on "computer science is fun" was higher for the students in the activity group than for the students in the research group in the pre-survey, but lower in the post-survey. These differences could have been caused by the fact that the research students were older, but we believe the main cause was the relatively little time that we were able to spend with the activity students. Perhaps more light is shed on the experiences of the activity students when we observe that on the post-survey every participant in the robot activity marked that they strongly agreed with the statements "computer science is challenging" and "computer science is useful", even though they did not feel this way at the beginning of the session.

By way of contrast, the students in the research group felt less strongly that "computer science is useful" after the four weeks. On the other hand, they felt much more strongly that computer science is fun. They also felt that they had become more capable programmers, indicating that they believed they acquired useful skills during the session.

Understanding of computer science

In looking at the responses to the question "What is computer science?", we saw two themes in the pre-survey: computer science as studying how computers work, and computer science as the study of using computers to help with science. In the post-survey several of the research students defined the field in terms of programming.

For the question "What do computer scientists do?", the pre-survey responses included "program", "no idea", "create computer parts", "experiment with science through computers", "hard work", and "learn and explore the different things computers can do and how they may be useful." In the post-survey the students all seemed to have more of an idea in their own mind of what computer scientists do, though not surprisingly their responses were heavily biased by their experiences in the program. Post-survey responses included: "research scientific materials, such as chemistry or robotics, by using software to virtually see results", "Find new things to teach computers and robots", "find new languages or work with how computers work", "they make new programs", "They look for more efficient ways to do things through programs given to robots", and our favorite "They have fun!"

Effect on college coursework

One of our goals in offering these activities was to introduce high school students to computing with the hope that they would both attend college, and take a CS class early in their college career. While the activity students are still in high school, the research students have now completed a few months of college. (100% of students in the PAYS program attend college; the vast majority enroll in 4-year baccalaureate-granting institutions.) We have heard from four of the six robot research students. Of those four, two are taking a CS1 class their first year in college, a third is planning to do so either later this year or sometime next year, and the fourth is looking into getting a CS class added to the course offerings at their institution. In addition, one of the five research students from the preceding year also took a CS1 class her first year in college.

DISCUSSION

While overall we feel we were successful in showing students why computer science is worth exploring, there are several things that we would do differently. For example, without being able to assign homework, it was difficult to bring the research students up to speed in terms of programming concepts and constructs, but almost impossible to do so with the activity students who saw us for only two hours a week. As a result, the activity students all strongly agreed with the statement "computer science is challenging" by the end of the four weeks. However, they also all strongly agreed with the statement that "computer science is useful" by the end of the session, which suggests to us that the activity was still a valuable experience for them. In the future it would be interesting to experiment with environments such as Alice (http://www.alice.org) or Scratch (http://scratch.mit.edu) in the activity, as this would eliminate some of the chaos caused by trying to work with the physical robots under tight time constraints.

The research students, on the other hand, expressed interest in doing more hands-on exploration with the robots. As one student wrote in the post-survey: "We should do more projects once all commands are taught. Trying different combinations is exciting."

Ultimately we see promise in both types of activities, as well as in the strategy of working within a larger outreach program, for exposing students to computing concepts and motivating them to consider taking CS1 in college. We also note that the activities can be led by college students who have only taken introductory computer science courses and, as a result, may be a way to keep college students involved in computer science at a point where a summer research project might not be feasible. Leading these activities can also strengthen the understanding of computing concepts for undergraduate students and perhaps help with retention at the college level.

ACKNOWLEDGMENTS

First and foremost, we would like to thank the PAYS research and activity students for making this such a fun and memorable experience for us. We would like to thank Matthew Bradley for helping us set up the hardware and software and the Norris foundation for giving us the resources to buy the Scribbler robots. We are also grateful to Maria Tucker and Sergio Marin for allowing us to work with the PAYS program. Finally, the last author was funded in part by the National Science Foundation under grant #CCF-0446604. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] Pomona College Draper Center for Community Partnerships: Educational Outreach Programs. http://www.pomona.edu/administration/draper-center/educational-outreach.aspx, retrieved January 19, 2011.

[2] Kumar, D., ed. Learning Computing with Robots. Fall 2008.

[3] Prashad, S., Muhammad, M. N., Gupta, M., Curriculum - IPRE Wiki. http://wiki.roboteducation.org, retrieved June 5, 2009.

[4] Expanding Underrepresented Minority Participation: America's Science and Technology Talent at the Crossroads. National Academies Press. http://www.nap.edu/catalog.php?record_id=12984, retrieved November 30, 2010.

[5] Peluso, E. M., Mauch, E., Incorporating Alice into a summer math and science outreach program. *Proceedings of the 2009 Alice Symposium*. 2009.

[6] Scarlatos, L., Lowes, S., Sklar, E., Chopra, S., Parson, S., Rudowsky, I., Holder, H, Building bridges: the 2006 summer institute, *J. Comput. Small Coll.* v23(3): 23-30, January, 2008.

[7] Das, M., Novel summer camp for underrepresented minority high school students, *Proceedings of the ASEE Conference*, Washington D.C., 1997.

[8] Freeman, P.A. and Cuny, J., Common Ground: A Diverse CS community benefits all of us, *Computing Research News*, v17(1), 2005.

[9] Broadening Participation in Computing Research and Education: Report of Computing Research Association Workshop on Broadening Participation, October 2004, http://www.cra.org/Activities/workshops/broadening.participation/report.html.

[10] Chen, T.-Y., Research mini-projects for high school students. Poster presented at SIGCSE 2009. Chattanooga TN, March 2009.