July 2014

Investigating Access to and Attitudes toward Programming in a Physics Camp

Gina Quan
University of Maryland at College Park

Ayush Gupta
University of Maryland, Baltimore

Recommended Citation
Gina Quan and Ayush Gupta. "Investigating Access to and Attitudes toward Programming in a Physics Camp" *Physics Education Research Topical Group, American Association of Physics Teachers* (2014).

Investigating Access to and Attitudes toward Programming in a Physics Camp

Gina M. Quan and Ayush Gupta

Department of Physics, University of Maryland, 082 Regents Drive College Park, MD 20742

Abstract: Computer programming has become a critical skill in much of physics research and undergraduate physics coursework. Our aim is to understand students' relationships (epistemological and affective associations) to coding and design, and in particular, how they experience and perceive access to programming in physics contexts. We piloted a project-based instructional module using Arduino Rovers (Arduino-integrated programmable robot-tanks) in a summer camp for high school students hosted by University of Maryland Physics Department. Throughout the program, participants worked through several open-ended design tasks before designing and completing a final project. In interviews, we asked students to reflect on their experiences programming and their perceptions of coding before and during the camp. Students in the program perceived different barriers to aspects of programming and design. These have implications for the roles students take up in activities of design and programming and whether they continue to seek such experiences in the future.

Keywords: undergraduate education, programming, Arduino, access, affect
PACS: 01.40.Fk, 01.40.G-, 01.40.ek

INTRODUCTION

Coding (programming) and design are important skills within the physical sciences. Physics often requires computational modeling, motivating the need for physics students to develop computational thinking [1-3]. The Next Generation Science Standards also identify engineering design and computational thinking as critical skills for K-12, and a need for students to learn engineering design practices alongside more traditional science practices [4].

Among education researchers, there is general consensus that student's early experiences are consequential to their continued and future participation in STEM fields. For example, Stevens et. al. followed engineering students' trajectories through their undergraduate years and found that students' successes in early experiences in engineering, even small ones, are consequential to their "becoming an engineer." These experiences color their perceptions of engineering and their self-concept as engineering students and future professional engineers [5]. While Stevens et. al. explored macro-dynamics of students' trajectories, Tonso investigated the micro-dynamics of students' socially constructed roles within classroom engineering design activities. The positioning of a student with respect to other members of their team, the tasks they undertake, and ultimately their experience in design was often strongly constrained by a rigid school culture; this influenced how centrally students participated in activities of design and were recognized for their contributions [6]. In a similar vein, Boaler and Greeno found that students' early experiences in mathematics led to their feeling identity misalignment and lack of agency, ultimately determining whether or not they persist [7]. Margolis and Fisher explored ways in which typical school culture misaligned with the ways women expressed their enthusiasm for programming. Campus culture often valued kinds of "geek" cultural practices produced by men, leading to attrition of women [8].

Integrating the findings from these lines of research would suggest that the specific roles that students take up within early design experiences are coupled to their success and whether they (and others) see them as central 'players,' which in turn could influence future access to opportunities and their persistence in engineering.

Providing positive early experiences within the physical sciences and programming may be a potential route to more women identifying with and pursuing these disciplines in the future. Margolis and Fisher recommend all-girl events and outreach to high schools as ways to spark and maintain women's interest in computing [8]. However, limited research is available on *mechanisms* by which such early experiences work (or don't), leaving it to instructors' intuition to make early experiences (such as summer camps, after school programs, etc.) more productive.

We investigate female students' sense of access to and attitudes toward coding while participating in a 2-week, targeted design experience within University of Maryland's Summer Girls program, which is described in the next section. We consider how a student's role within her team couples with changes in how she relates to physics. In this paper, we focus on two two-member design teams where we see changes in participants' perceptions of access and attitudes. We aim to unpack these changes, given our limited data set. These analyses contribute to a broader understanding of *how* access to programming may be shaped by early experiences.

CLASSROOM BACKGROUND

The study focuses on high school students in Summer Girls, an all-expenses-paid 2-week summer outreach program sponsored by the UMD Physics Department and Joint Quantum Institute. Rising juniors and seniors complete a short application and are selected to create a cohort of students who share an interest in science, hold a diverse set of experiences, and could benefit from participating in physics camp. The program included a design component (designed by the first author), in which students learned to program Arduino (microcontroller) controlled robot-tanks (henceforth, Arduino-bot). The Arduino-bots had motion functionality and were fitted with a basic distance sensor. Students spent the first week of the program working 1-2 hours per day on self-paced design tasks in pairs. Design tasks required students to program the Arduino-bot to perform some task such as detecting an obstacle, visually depicting distance from a wall, etc. In the second week of the program, students designed and implemented final projects in pairs and trios. Students were provided with a variety of resources, including a library of sample code. The classroom was staffed by two instructors and two to three volunteers each session. We dedicated roughly twenty hours of the entire camp to design activities. The rest of the camp featured modern physics lectures, lab tours and activities. One lab tour guide emphasized the use of Arduino in data collection, while another tour guide related similar devices in their lab to the Arduino projects.

Students did not know ahead of time that programming and design will be part of the program, and we may have captured students who would otherwise shy away from programming or not see programming as a relevant skill to physics.

METHODOLOGY

Our data comes from videotaped in-class group-work and interviews during the 2013 program. Due to limited resources on the project, we collected video records of one design team during their design activities and interviewed 4 students as part of the program evaluation. Students were interviewed with their final project partners. Surveys about students' attitudes toward physics and coursework were also collected as part of ongoing program evaluation. The interview probed student interest and engagement in physics and aspects of the camp. Students were asked to describe their experiences in the Arduino component.

We watched interviews to identify students' roles within the design/programming team and their attitudes toward the Arduino component of the program. Initial passes through the data focused on moments where students demonstrated strong affective response. Interviews were then transcribed and analyzed. In analyzing these interviews, we focused on identifying aspects of the program that students felt barriers or access to, students' sense of barriers shifting, and what activities students participated in that contributed to those shifts. The analysis was presented at multiple seminars to other research team members to ensure the representation of multiple interpretations of the data and assistance with sorting out the interpretation supported by the largest fraction of the data.

Next, we present data excerpts from two pairs of students, which highlight access to programming.

HAZEL AND OLIVE: POSITIONAL DYNAMICS AND ROLES

The example of Hazel and Olive problematizes simplistic stories of the relation between role-taking and students' self-concept with respect to programming. In their group, Hazel took the lead on the programming tasks while Olive focused on mechanical tasks. For example, Hazel singlehandedly programmed a "get unstuck" program for the Arduino-bot while Olive built a mechanical arm for the robot to throw confetti. Hazel and Olive were aware of this difference in their roles and, in their interview, tied this to their relative comfort with programming:

> Olive: It was really difficult [to learn coding] because I didn't actually have no idea what we were doing. And so I was glad to have Hazel as a partner because she helped me.
> Hazel: I hope I explained /Olive: You did, you did explain/[1] what I was doing.
> Hazel: Cause I just went off on my own, a lot, /Olive: Yeah she did, well- / I felt bad about that. I was like oh no, I totally forgot she was there.
> Olive: Yeah um, without her helping me, I actually had no idea what we were doing.

In this segment of interaction, Hazel is positioned by both Hazel and Olive as the programming expert who

[1] / Indicates overlapping speech

almost singlehandedly executed the programming tasks, while Olive is positioned as the novice whose programming contributions were minimized. Their differential access to the programming task in the summer program was consistent with their previous experience with programing: Hazel had a substantial programming background that she described as contributing to her comfort with working with the Arduino microprocessor, whereas Olive had no previous experience and expressed that she found learning programming to be difficult.

On the one hand, we might think that Olive's membership on a cohesive team that did lots of programming would unproblematically help her gain access to programming. On the other hand, we might think that Olive's positioning as a complete novice, with Hazel as the expert who actually did all the programming, would deny Olive access to programming, taking away from Olive valuable opportunities to engage in learning to program at a slower pace, make mistakes, and learn from her mistakes. We see that the situation is more nuanced than either of these "extreme" narratives. Within the earlier quotation we note the harmonious student-tutor positioning between Hazel and Olive that positively contributed to Olive's learning and to her relationship to programming in the future. Later in the interview, she remarks "I had no idea what coding was… And something new that I didn't think I'd be open to, and now that I'm learning more about it, I want to learn more, like have more insight and know more about it cause it's something I obviously didn't know. And she was a good explainer...I think she was probably the best one to explain it to me."

Through her interaction with Hazel, Olive gained access to coding in the Arduino-bot environment, something that was new to her and motivating her to learn more in the future.

ROSE AND SCARLET: CHANGING PERCEPTIONS OF BOUNDARIES

The "same" experience within a course can influence different students (even within a single team) very differently, depending on their past experiences and their roles within the team. From another design team, Rose gains what the Summer Girls program hoped she would - a better grasp of particular skills and ways of thinking; while her partner, Scarlet, values the experience of getting to try new things and prove to herself that she can learn in new ways.

Rose was a student for whom participating in the Summer Girls, specifically the design task, was transformational in helping her feel closer and more comfortable with programming and boosting her desire to study it in college:

> Rose: I had always thought of coding and computers as something away from me that I didn't have access to, that I probably wouldn't get to until college...Now I'm interested, I really wanna learn more, and even if it's not- even if I don't get a chance before college, at least I know in college I wanna take some programming classes... Umm, probably because in my school… we have, I think, computer science, but I was never aware of these things as being accessible to anybody who didn't have any experience. So, the first time I heard about AP Comp Sci was when my friend was taking it, and she's you know, good with computers so I just assumed you have to, like, know a lot before you take it. But, so it wasn't really something that I had- like I hadn't gotten onto that track, so I felt like I wouldn't get onto it... [now] I feel like I could go into it and have interest and be able to learn.

To Rose, programming was almost another world that she could see but not enter. The summer program provided Rose with successful early experiences that bent her perceived trajectory.

Scarlet, on the other hand, sees programming as a world that she knew little about and which the Summer Girls program enabled her to experience. Scarlet frames this experience, however, not in terms of programming but at a more meta-level: Summer Girls was a place where she became comfortable trying new things, which for her constituted learning in a new way (getting "just thrown into it").

> Scarlet: Umm, I feel like when I first got here, I was a little bit - just kinda freaked out. There's SO many things /Rose: Yeah/ that I don't know how to do and I'm just thrown into it… Something consistent like the Arduino project got me comfortable with kinda trying out new things and as we did so many different other things I got more and more comfortable doing that and I had a lot of fun.

Later in the interview she said:

> I was really surprised by the coding. I had no idea that we were going to do that but I'm glad that I did...I had no idea that you could use the computer, I thought it was just some machine that does it all for you. But it's actually you, like telling it to do things which I thought was really cool. But it's kindof - I knew absolutely nothing about it, and now I feel like I have a good general knowledge about it which is fun, and I think I'd wanna learn more about it.

Before, Scarlet felt uncomfortable trying new things and had little experience with programming, but she now suggests that she would be interested in pursuing further programming experiences. This stance towards trying new things is also a part of expert design thinking [9]. Epistemologically, she comes to see the computer as a tool that she can use towards her design objectives, a stance that is also laden with affect ("really cool").

DISCUSSION

This work sheds light on how students experience programming and design, and how those experiences contribute to their perceptions of their future participation in programming. Hazel and Olive took on siloed roles in their problem solving, which led to Olive gaining access to the role of a student-coder. Olive's positioning as the novice, in contrast to Hazel as the expert programmer, supported Olive in feeling that she could learn programming. We also see Scarlet gain an affective shift toward trying new things. Olive, Rose, and Scarlet all express a sense of greater accessibility toward programming going forward. That Rose would not have sought out programming on her own suggests the importance of making these disciplines easily available to students.

Informal education experiences such as Summer Girls are often created with the intention of developing and nurturing science interests. Evaluation of the success of these programs then should also consider how students' sense of access changes as a result of the program. Here, however, we do not explore how perceptions of access toward programming or participation in the program has altered perception of access to science.

This manuscript presents work in progress, leaving much unanswered: what material, instructional, and social features contributed to Olive feeling comfortable learning from Hazel? How were members of the same collaborative team able to have such different experiences? Pairing classroom observations with interview data could yield greater insight into these questions.

A better understanding of key classroom features will also help instructors in designing for more productive design/programming experience for students. But we can speculate some instructional conclusions based on the current data.

We can imagine instances where similar student pairings as Hazel-Olive or Rose-Scarlet would play out differently. Olive could have felt shut out of programming, and her lack of programming could have become a weakness of the team. We can also imagine Olive taking on central tasks of programming in a desire to keep up with the more expert partner. The multiple paths available to groups and group-members with similar initial conditions suggests that there might not be fixed instructional implications, rather, instructors of design teams need to attend and respond to individual team dynamics. These kinds of relationships can't necessarily be engineered for (i.e. pairing a novice with an expert so the novice can gain access), and so it is important to support teams in being productive in their own ways.

Rose and Scarlet, both with little prior experience in programming came out of the program wanting to learn more programming. But there are nuanced differences in what they express as having gained through participation in Summer Girls; in particular, Scarlet's expressed comfort in trying out new things, something that wasn't an intentional learning goal for the instructors. This illustrates the importance of recognizing what students perceive to be successes – beyond the instructional goals articulated a priori.

ACKNOWLEDGMENTS

We thank Andrew Elby, Vashti Sawtelle and Stephen Secules for comments on earlier drafts of this paper. We also thank the UMD PERG, Science Education, and Engineering Education group members for discussions around the presented data. This work is supported by DUE- 1245590 and the UMD Physics Department.

REFERENCES

1. J. M. Aiken, et. al, in *Physics Education Research Conference Proc.* **1513**, 46-49 (2013).
2. P. Sengupta, J. S. Kinnebrew, S. Basu, G. Biswas, and D. Clark, *Education and Information Technologies* **18** (2), 351-380 (2013).
3. B. L. Sherin, *Intl. Journal of Computers for Mathematical Learning* **6** (1), 1-61 (2001).
4. R. Carr, L. D. Bennett, and J. Strobel, *Journal of Engineering Education* **101** (3) (2012).
5. R. Stevens, K. O'Connor, L. Garrison, A. Jocuns, and D. M. Amos, *Journal of Engineering Education* **97** (3), 355-368 (2008).
6. K. Tonso, *Journal of Engineering Education* **95** (1), 25-37 (2006).
7. J. Boaler and J. G. Greeno *Multiple Perspectives on Mathematics Teaching and Learning* 171-200 (2000).
8. J. Margolis and A. Fischer *Unlocking the Clubhouse: Women in Computing* (MIT Press, Cambridge, 2003)
9. T. Brown, *Harvard business review* **86** (6), (2008).