

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/360682981>

Recruiting K-12 Youth into Computer Science: Insights from a Multi-Year Research Project

Article in ACM Inroads · June 2022

DOI: 10.1145/3524916

CITATIONS

0

READS

207

2 authors:



Chongning Sun

Indiana University Bloomington

11 PUBLICATIONS 135 CITATIONS

SEE PROFILE



Jody Clarke-Midura

Utah State University

84 PUBLICATIONS 2,095 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Virtual Performance Assessment Project [View project](#)

Recruiting K-12 Youth into Computer Science: Insights from a Multi-Year Research Project

By Chogning Sun, *University Information Technology Services*, and
Jody Clarke-Midura, *Utah State University*

According to a recent report by the US National Science Board [38], Computer Science (CS) has the second lowest participation rate for women of all occupations. Despite over thirty years' efforts in broadening participation, why are only a small number of women involved in CS? As educators, what can we do to broaden participation? In this article, we share insights gained from research—both ours and others'—and the strategies that may be of use in encouraging women's participation in CS.

INTRODUCTION

Despite over thirty years of research on broadening participation in Computer Science (CS), women are still underrepresented in CS [26]. Research has identified a number of factors for why women are not entering CS, such as: environments that are perceived as not being inclusive to women [36,37], lack of support [1,52], lack of role models [10,21], low interest [25], low self-efficacy, which is defined as an individual's belief in their ability to do a task [3,8,41], and negative attitudes toward computing [27]. Research has further identified factors for why women pursue CS: social support, opportunities to program, knowledge about CS, and self-efficacy [52]. Similarly, Social Cognitive Career Theory (SCCT), a theory that is used to study career interest and choice of career, identified self-efficacy as a precursor to interest and a factor in career choice [33,34]. Giv-

en the importance of self-efficacy in women's decision to enter CS and its role as a precursor to interest, we decided to design a program around increasing self-efficacy in CS, that would hopefully lead to interest in CS, for girls.

We designed a summer camp program where we teach youth to design and code mobile phone apps using MIT App Inventor, a block-based programming language. While our focus was on girls, we designed the App camp program to be inclusive and equitable for all learners based on research and strategies for broadening participation in CS. Our goal was to reduce some of the barriers that prevent youth, especially girls, from entering CS. In the following sections, we describe our program design and findings from our research over the past three years. Specifically, we discuss the curriculum, learning activities, our near-peer mentoring/role modeling approach, and other strategies such as using student-created apps as an empowering tool in order to increase youth's interest and self-efficacy in CS. Our goal in this document is to share research-based strategies that others can use to design informal CS programs for youth, girls in particular.

THE APP CAMP PROGRAM

We have been running the App camp program in a rural area in the Intermountain West of the U.S since 2017. From 2017 to 2019, we offered fifteen week-long camps, five camps per summer, which included one mentor training camp (6 hours/day,

30 hours total) and four App camps (3 hours/day, 15 hours total each). Over the mentoring training camp, mentors were trained on how to be a mentor and code eleven mobile phone apps using MIT App Inventor. They returned a week or two later to mentor the campers to code the same apps.

We recruited the mentors from the attendants of our previous camps, referral of previous mentors, and local high schools. The applicants had to complete an online application form and then went through a follow-up interview process, where they were inquired about their previous programming and mentoring experiences, and their approaches to different scenarios while mentoring. Although we asked all applicants about their previous programming experiences, it was not a requirement. Forty-three percent of the mentors we hired did not have any programming experience before the app camp. We hired more females than males with the intent to increase the proportion of female role models in our mentor pool. The mentors were rewarded with a stipend of \$200 for completing the training session and mentoring one App camp session. They were assigned to the App camp sessions based on their availability.

We used flyers, social media, local publications, and school email systems to recruit the campers, charging a nominal camp registration fee to encourage commitment for signing up and waiving the fee for anyone who requested it.

Over the three years (i.e., 2017–2019), the App camp program reached sixty-eight high-school-aged mentors (female = 48, male = 20, average age = 15) and 279 middle-school-aged campers (female = 166, male = 113, average age = 12). Three percent of the mentors self-identified as LatinX. Mentor racial makeup included 12% Asian, 82% White, and 6% Multi-racial. Due to lack of parental consent, four campers' demographical information and research data were missing and thus removed

from analysis. Of 275 campers, 3% self-identified as LatinX. Six percent campers reported to be Asian, 1% as African American, 3% as Native American, 76% as White, 8% as Multi-racial, and 5% as Other. Please note that while we strove to recruit youth of both genders and different ethnic/racial backgrounds, the low representation of African American youth in our sample reflects the actual racial makeup in the state's population (approximately 1.5% in 2021) [49]. Twenty-nine percent of the mentors and 18% of the campers were on free or reduced lunch.

In the following section, we discuss our program design as well as its theoretical groundings.

DESIGN CONJECTURE 1: USING A NOVICE-FRIENDLY CODING ENVIRONMENT

Research on visual block-based programming environments suggests they lower the barriers of initial entry to programming [42,50]. Although these languages use the drag-and-drop technique that allows the coder to drag the blocks that contain the programming command and then piece them together with other blocks, they are capable of introducing complex programming concepts [51], but also powerful enough to allow for originality and creativity [39]. In our design, we selected MIT App Inventor, a block-based language developed to create mobile apps. Figure 1 presents an example of the interfaces of App Inventor—the right side shows the app interface while the left side shows the codes. On the right shows the app interface, while on the left shows the codes. In addition to being friendlier to novice programmers, MIT App inventor allows users to download what they create (i.e., apps) to a mobile device and share with other people, something we encouraged throughout all of our App camps.

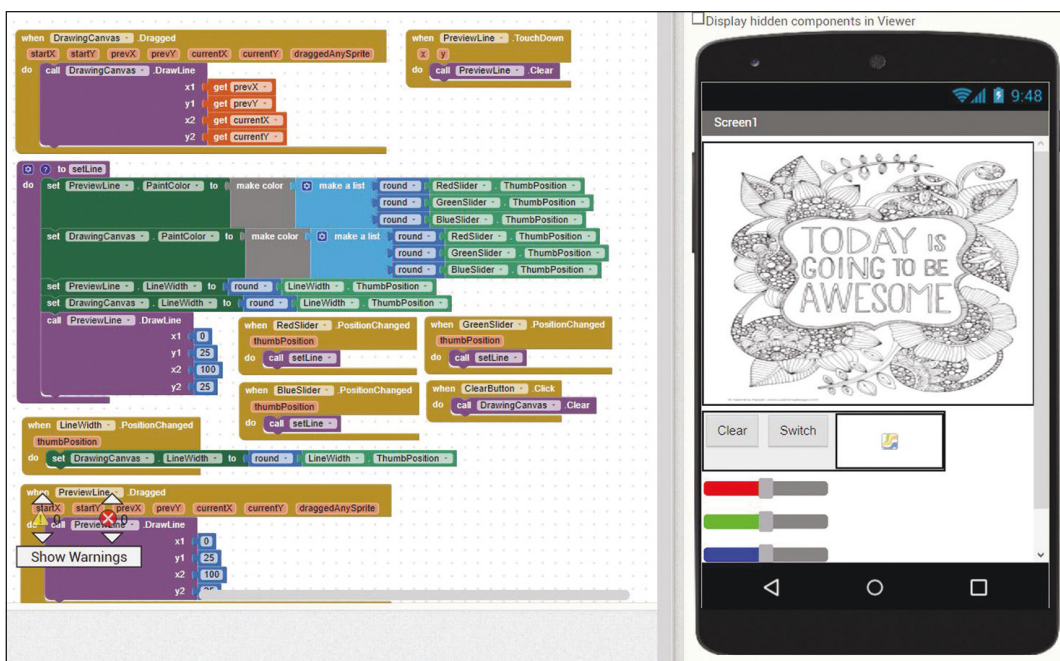


Figure 1: An Example of the Interfaces of App Inventor.

**DESIGN CONJECTURE 2:
DESIGNING CODING TASKS THAT SPEAK
TO YOUTH'S INTEREST**

As smartphones become more accessible and apps become a popular venue for creative expression, the act of designing and coding mobile apps has the potential to attract and interest youth in CS [50]. Interest can take time to develop; however, creating apps that may have a novelty effect (i.e., a positive effect brought by a brand new design experience and more importantly by a new design tool) has the potential to spark initial interest for youth [18,29]. Therefore, in our camps we use apps to spark youth's interest and pull them in and then work on helping them develop self-efficacy. The main learning activity in our camp was to program up to 11 apps.

Furthermore, research shows that one reason girls avoid CS is because computing is considered to be masculine and geeky and does not resonate with girls' personal interests [2]. Therefore, we centered our curriculum design around the youths' personal interests by allowing them to personalize and customize their apps. To do so, we only showed the necessary steps and components for creating a project while intentionally not showing the final solution in order for youth to repurpose each app in solving a different problem of their design. Additionally, we asked our near-peer mentors in the camps to showcase the apps they created to give campers ideas of how to tinker with the codes. Figure 2 presents an example of campers coding their apps.

**DESIGN CONJECTURE 3:
MANAGING LEVELS OF TASK DIFFICULTY**

We purposefully designed the sequence of the activities and level of difficulty. Designing tasks that slowly increase in difficulty and presenting challenges are effective ways to promote self-efficacy and interest [5–7,20,29]. For this reason, our curriculum started with simple apps that provided detailed instructions in a step-by-step format accompanied with images. Each app built upon CS concepts introduced in earlier apps while also introducing new concepts. As such, instructions for previously addressed concepts were not provided when those concepts were used again. As the scaffolds were removed, the campers were encouraged to ask questions and problem solve amongst themselves and with their near-peer mentors. We also provided free-build opportunities for advanced campers to challenge themselves, where we presented specific problems and asked campers to design and program their own apps to solve the problems.

**DESIGN CONJECTURE 4:
INTEGRATING MENTORS AS ROLE
MODELS**

Mentoring and role modeling are commonly used strategies in informal and formal CS education to recruit and retain women and underrepresented students [2,17,31,47]. Relatability and similarity to mentors/role models can have a positive impact on mentees' affect [5,6,40,43,48]. There is also evidence that sug-



Figure 2: Campers Coding their Apps.

gests mentors with a high level of expertise may have a negative effect on mentees' affect [4,30]. For example, experts may not be relatable such that a mentee may think they will never be able to aspire to that level of expertise [44].

We provided two types of role models: virtual models (video stories of people who are slightly older yet with similar backgrounds talking about how they became interested in CS and what they do in their current CS jobs), and live role models in the form of near-peer mentors who were similar to the mentees in terms of age and coding skills (see Figure 3). In order to increase their role modeling effect, we designed a near-peer mentoring model based around the sources of information that promote self-efficacy [5–7]. See [45] for details on the near-peer mentoring model and its configurations. Based on the conceptual near-peer mentoring model, we developed a training curriculum to train mentors to perform three practices when they worked with the mentees: (a) providing modeling, (b) providing encouragement, and (c) using questions to guide campers' thinking rather than just telling them how to program their apps.



Figure 3: Near-peer Mentors and Students.

DESIGN CONJECTURE 5: SHOWCASING YOUTH CREATED ARTIFACTS TO ELICIT SOCIAL SUPPORT

Social support, especially parental support, is important to girls' engagement in CS [1,52]. While research shows parents usually have a higher expectation about boys' success in STEM than girls' [35], sharing with parents the artifacts they create is an effective way to register parents' attention to their daughter's potential and enlist their support in regard to choosing a STEM pathway [32]. In addition, the lack of support is sometimes ascribed to parents' limited knowledge of the CS field and the associated opportunities, especially in ethnic-racially and/or economically disadvantaged communities [28]. Showcasing child-created artifacts can be an exposure opportunity for the parents to CS and can help them see the usefulness of CS. Research shows that

We provided two types of role models: virtual models (video stories of people who are slightly older yet with similar backgrounds talking about how they became interested in CS and what they do in their current CS jobs), and live role models in the form of near-peer mentors who were similar to the mentees in terms of age and coding skills ...

parental career-related attitudes also influence children's career choices [19]. For example, when parents value an occupational choice, children are more likely to value it, too [22,23]. During the camp, we encouraged the campers to take the devices home and share their apps with their family and friends.

MEASURING THE APP CAMP EFFECTS

We collected both quantitative and qualitative data to measure the App camp effects on youth's affective attitudes (i.e., self-efficacy and interest) toward computer programming. In order to measure those affective changes, we administered a pre- and post-test using a multi-scale, multi-item affect survey on the first and last day of the camp. The affect survey included measures about perceived father and mother support, self-efficacy, interest, and value beliefs about the utility of computer programming (i.e., utility value). These measures were adapted from [9,23,24].

As the program progressed, we developed more scales and added them to the post-camp affect survey to measure campers' camp experiences (e.g., task difficulty, comfort level working with mentors) and their perceptions of the near-peer mentors' mentoring practices (e.g., act of role modeling, providing encouragement). These new measures and the affect measures were written on an 8-point Likert scale (1 = Strongly Disagree, 8 = Strongly Agree). We also added items to investigate whether campers took their devices home and shared the apps they created during the camp and with whom. In order to explain how the App camp activities affected campers' self-efficacy and interest, we designed and conducted a series of exit interviews and camp observations over the years, with each summer focusing on one or two areas of investigation that were informed by the previous quantitative findings. For example, we started with an exploration of what camp design features contributed to camper affective changes and what social support they had in relation to computer programming. We then proceeded in the following summers to focus on the near-peer mentors and how they affected the campers' self-efficacy and interest.

FINDINGS

In this section, we present a brief summary of findings of our research on the App camp program. As a reminder, the data we used were quantitative and/or qualitative data collected between 2017 and 2019. A major and consistent finding was that campers' self-efficacy and interest in computer programming significantly increased after attending the camp as measured by pre- and post-surveys. This trend was observed across three different cohorts of campers across the three summers [11,13,14,16,45,46], and also when we combined the three years data and only looked at the girl camper data [15]. While we used different indices to measure the magnitudes of changes in self-efficacy and interest in [11–14,45,46], we generally observed medium to large effect sizes for self-efficacy and small effect sizes for interest. For girl campers, there showed a large effect size of increase in self-efficacy, Cohen's $d = .88$ and a medium effect size of increase in interest, Cohen's $d = .53$ [15]. *This suggests that our design is effective in promoting youths' (girls' specifically) self-efficacy and interest and has the potential to broaden participation in CS.*

Next, one of our studies [12] found that perceived father and mother support both predicted campers' utility value, which predicted interest in computer programming. In other words, parents' support in their children's pursuit of CS affects children's beliefs about the usefulness of CS, and these beliefs affect to what extent they are interested in CS. Our statistical analyses further showed that mother support also directly influenced children's interest, while father support did not. These findings suggest the importance of parental support, especially maternal support, in recruiting youth to CS.

In the same study, we found that perceived maternal support significantly increased from pre- to post-camp, $r = .23$, indicating a small to medium effect size (we conducted a nonparametric test and thus used r as the index of effect size), while father support did not show statistically significant change. This means during the camp, campers perceived that their mothers became more invested in their pursuit of CS. Interview data showed that the positive change in perceived maternal support was partly owing to the fact that mothers showed great interest in the apps their children made and encouraged them to learn more about computer programming.

In a follow-up study using another cohort of campers [16], we encouraged campers to take their device (i.e., project-provided smart-phones) home and share their apps with their family. We also found a significant increase in perceived maternal support after the App camp, which showed a medium effect size, $r = .32$. However, different from our previous findings [12], this study found that perceived paternal support also increased

significantly, $r = .28$, medium effect size. Interview data showed that taking their device home played an important role in promoting campers' perceived changes in parental support. Sharing the apps with family not only increased the parent-child interactions around the camper made apps (e.g., parent(s) testing out the apps with child and/or child explaining to parent(s) how they coded the apps), but also led to conversations about the importance of CS and/or pursuing a CS career. According to the campers, these experiences made them feel more confident in their programming skills and interested in learning more about computer programming. *Findings from these two studies suggest that asking campers to show and tell their apps to their family was effective in eliciting parental support, which plays an important role in recruiting youth to CS.*

In addition to findings about parental support, the [16] study also identified other design features of the App camp as contributing to the growth of camper interest. Specifically, camper interviews produced evidence showing that *watching the video stories of CS role models*

increased campers' interest in CS careers by showing them what a CS career looks like and that the goal of seeking CS careers or skills is attainable. Being able to personalize the apps was another reason campers mentioned in the interviews that made programming interesting. We saw a similar finding in another study [45], where *interview data provided additional evidence in support of the effectiveness of the app personalization strategy in promoting interest*. For example, to explain what made coding the apps interesting, one camper stated, "You are kind of free to put your own spin on stuff. Like put your own. Like you put yourself into it. Like you put your personality to it."

Since a critical component of our camp design was the employment of near-peer mentors as role models, we conducted several studies to explore the predictive relationship between the near-peer mentorship and camper affect (i.e., self-efficacy and interest) [11,45,46]. First, this just-cited research showed that the role modeling effect that the near-peer mentors embodied predicted camper self-efficacy, meaning that *the near-peer mentors served as role models to the campers, which increased the latter's self-efficacy in performing computer programming*.

In order to understand what factors influenced the role modeling effects, we focused on mentors' personal attributes and mentoring practices. For example, in [46], we tested the relationship between campers' perceived similarity to a mentor regarding programming skills (i.e., perceived competence similarity) and mentor role modeling effect on camper self-efficacy. We detected a significant interaction effect between perceived similarity and role modeling. More specifically, when a mentor

Findings from these two studies suggest that asking campers to show and tell their apps to their family was effective in eliciting parental support, which plays an important role in recruiting youth to CS.

embodied a strong role modeling effect (i.e., one standard deviation above the mean of role modeling scores), perceived similarity increased the effect of role modeling on camper self-efficacy—holding role modeling constant, the more similar a mentor was perceived to be to a camper in terms of programming skills, the more greatly camper self-efficacy increased. On the other hand, when the role modeling effect was weak (i.e., one standard deviation below the mean of role modeling), perceived similarity did not play a significant role in affecting role modeling on self-efficacy. In other words, when a mentor only embodied a weak role modeling effect, whether they were perceived as similar or not, campers' gains in self-efficacy did not differ substantially. To sum up, *this finding suggests that perceived competence similarity is an important factor for consideration when it comes to role modeling.*

In another study [13], we identified through interview data that being relatable and similar in regard to age, personality, personal interests, programming experience and skill level was associated with the credibility of the near-peer mentors as role models. We further found through statistical analysis that mentor relatability (which we defined as a sense of similarity and affective connection to the mentor) predicted camper self-efficacy and interest. To state it differently, if a camper felt related to the mentor, their self-efficacy and interest were more likely to increase. Interview data indicated that the sense of feeling related to their mentors helped campers see the mentors' achievements in CS as more attainable, which in turn promoted their self-efficacy and interest in doing the same.

As reported in a follow-up study [11], we found the same predictive relationship between mentor relatability, self-efficacy, and interest—that is, mentor relatability predicted both self-efficacy and interest. Apart from that, we also found that (a) both mentor relatability and role modeling predicted interest, and (b) similar to [46], mentor relatability moderated the role modeling effect (i.e., there was an interaction effect between mentor relatability and role modeling), but on interest. More specifically, when mentor relatability was strong (i.e., one standard deviation above the mean), role modeling did not significantly affect campers' interest. However, role modeling had a significant impact on campers' interest at lower levels of relatability. Consistent with others' work [40], our research provided strong evidence on the relationship between the quality of being relatable and motivation development. Observation

and interview data collected in the same study showed that the practices of sharing personal stories (e.g., mentors shared their own coding struggles with campers) and making connections (e.g., mentor showed interest in campers' out-of-camp life such as school life) were associated with mentor relatability, while showcasing mentor-created artifacts (e.g., mentor showcased their apps before campers started to code them) and mentor as

role models (e.g., campers reported to be inspired by their mentors and wanted to emulate them in terms of computer programming) contributed to role modeling.

In a mixed-methods study [45], we investigated how the near-peer mentors' mentoring practices influenced campers' self-efficacy. First, quantitative and qualitative findings converged to show a positive change in campers' self-efficacy from pre- to post-camp, Cohen's $d = .73$, indicating a medium to large effect size. Further, statistical analysis showed that camper gender was not a significant predictor of gains of self-efficacy, meaning that the magnitudes of gains in girl and boy campers' self-efficacy did not

differ significantly. Second, we found that mentor encouragement directly predicted campers' changes in self-efficacy, while despite no direct influence on self-efficacy, behavioral modeling (a sub-process of role modeling) and instructive feedback (i.e., mentor verbal feedback that served the purposes of content instruction, problem-solving, and performance assessment) enhanced campers' task performance, which promoted their self-efficacy in programming. Qualitative findings corroborated the quantitative findings concerning the predictive relationship between the three mentoring practices (i.e., behavioral modeling, instructive feedback, and providing encouragement) and camper self-efficacy. Additionally, the interview data also provided insights into how contextual factors such as task difficulty and personal factors such as campers' previous knowledge may have influenced the effectiveness of the near-peer mentoring model. For example, many campers we interviewed commented that the programming tasks were not that difficult to them, and many had previous programming experience. These factors were likely to have neutralized the impact of mentors' act of modeling on some campers' efficacious beliefs. Additionally, we did not find qualitative evidence suggesting that the near-peer mentoring model affected self-efficacy differently as a factor of camper gender. *The preliminary evidence we found from [11,45,46] suggests the potential of using*

Based on our research and experience, we recommend the following practices and strategies to anyone who is interested in promoting inclusion and equity in CS informal settings ... Providing a beginner-friendly coding environment ... Designing curriculum and learning activities that leverage personal interest ... Integrating near-peer role models ... Providing opportunities to showcase youth-created artifacts.

Recruiting K-12 Youth into Computer Science: Insights from a Multi-Year Research Project

near-peers as role models to recruit youth to CS, and, more importantly, it sheds light on the possible underlying mechanism for why the near-peer mentoring model works (e.g., the relationships among mentor personal attributes, mentoring practices, and self-efficacy and/or interest).

We want to make a final note about our research, specifically, qualitative findings. While we constantly had our bias (i.e., subjectivity and positionality) in check and applied techniques such as double coding in the process of analyzing and interpreting qualitative data, researcher bias could not be utterly eliminated, which may have biased the qualitative findings.

RECOMMENDATIONS

Based on our research and experience, we recommend the following practices and strategies to anyone who is interested in promoting inclusion and equity in CS informal settings.

- *Providing a beginner-friendly coding environment:* Choosing a coding environment that has a low threshold such as a visual block-based language is beneficial to students who are new to programming, because such a tool can boost confidence.
- *Designing curriculum and learning activities that leverage personal interest:* Focus on connecting activities or coursework with youth's personal interests. Encourage youth to demonstrate their voice and personality in their work/creations. Leverage the novelty effect in your course design to spark students' situational interest; use challenges to maintain their interest and boost their confidence if they manage to tackle the challenges on their own.
- *Integrating near-peer role models:* Role models do not have to be someone who has a deep knowledge of the field; rather, they can be a fellow student who is relatable and happens to have more experience/knowledge. Additionally, we want to note that in-depth preparation of the mentors with both content and mentoring/teaching strategies are important to the success of workshops like these.
- *Providing opportunities to showcase youth-created artifacts:* Educators should make use of student-created artifacts to recruit parents' support and engagement in what their children are doing. Also, provide youth with the opportunity to share and showcase their works in order to increase their self-efficacy and interest. ♦

Acknowledgements

This study is supported by the U.S. National Science Foundation under Grant #1614849. Any opinions, findings, and conclusions expressed in this study are those of the authors and do not necessarily reflect the views of the National Science Foundation or Utah State University.

References

1. Alshahrani, A., Ross, I. and Wood, M.I. Using social cognitive career theory to understand why students choose to study computer science. in *Proceedings of the 2018 ACM Conference on International Computing Education Research*. (Espoo, Finland, 2018), 205–214.
2. Ashcraft, C., Eger, E. and Friend, M. Girls in IT: *The facts*. (National Center for Women and Information Technology, 2012).
3. Baker, L.J., Snow, E., Garvin-Doxas, K. and Weston, T. Recruiting middle school girls into IT: Data on girls' perceptions and experiences from a mixed-demographic group. in *Women and Information Technology: Research on Underrepresentation*, edited by J.M. Cohoon and W. Aspray (MA: The MIT Press, 2006), 115–135.
4. Bamberger, Y.M. Encouraging girls into science and technology with feminine role model: Does this work? *Journal of Science Education and Technology*, 23, 4 (2014), 549–561.
5. Bandura, A. Self-efficacy: Toward a unifying theory of behavioral change. *Psychological Review*, 84, 2 (1977), 191–215.
6. Bandura, A. *Social foundations of thought and action: A social cognitive theory*. (NJ: Prentice-Hall, 1986).
7. Bandura, A. *Self-efficacy: The exercise of control*. (W.H. Freeman and Company, 1997).
8. Blaney, J.M. and Stout, J.G. Examining the relationship between introductory computing course experiences, self-efficacy, and belonging among first-generation college women. in *Proceedings of the Conference on Integrating Technology into Computer Science Education*. (Seattle, Washington, USA, 2017), 69–74.
9. Carrico, C. and Tendhar, C. The use of the social cognitive career theory to predict engineering students' motivation in the produced program. in *ASEE Annual Conference and Exposition*. (San Antonio, TX, 2012), 25.1354.1–25.1354.13.
10. Cheryan, S., Drury, B.J. and Vichayapai, M. Enduring influence of stereotypical computer science role models on women's academic aspirations. *Psychology of Women Quarterly*, 37, 1 (2013), 72–79.
11. Clarke-Midura, J., Sun, C., Pantic, K., Poole, F., Feldon, D., and Allan, V. Design of a near-peer mentoring model to foster self-efficacy and interest in Computer Science. (in review).
12. Clarke-Midura, J., Poole, F., Pantic, K., Hamilton, M., Sun, C., and Allan, V. How mother and father support affect youths' interest in computer science. in *ICER 2018. Proceedings of the 2018 ACM Conference on International Computing Education Research* (Espoo, Finland, 2018), 215–222.
13. Clarke-Midura, J., Poole, F., Pantic, K., Hamilton, M., Sun, C., and Allan, V. How near peer mentoring affects middle school mentees. in *Proceedings of the 49th ACM Technical Symposium on Computer Science Education* (Baltimore, Maryland, USA, 2018), 664–669.
14. Clarke-Midura, J., Sun, C., and Pantic, K. Making apps: An approach to recruiting youth to computer science. *ACM Transactions on Computing Education*, 20, 4 (2020), Article 30.
15. Clarke-Midura, J., Sun, C., Pantic, K., Feldon, D., and Allan, V. The influence of gender composition in computer science camps on girls' self-efficacy and interest. in *AERA Annual Meeting [Roundtable session]* (San Francisco, CA., 2020), <http://tinyurl.com/vvxxm9p>. Accessed 2022 April 13.
16. Clarke-Midura, J., Sun, C., Pantic, K., Poole, F., and Allan, V. Using informed design in informal computer science programs to increase youths' interest, self-efficacy, and perceptions of parental support. *ACM Transactions on Computing Education*, 19, 4 (2019), Article 37.
17. D'Souza, D., Hamilton, M., Harland, J., Muir, P., Thevathayan, C. and Walker, C. Transforming learning of programming: A mentoring project. in *Tenth Australasian Computing Education Conference*. (Wollongong, Australia, 2008), 75–84.
18. Deci, E.L. and Ryan, R.M. *Intrinsic motivation and self-determination in human behavior*. (NY: Plenum, 1985).
19. Dietrich, J. and Kracke, B. Career-specific parental behaviors in adolescents' development. *Journal of Vocational Behavior*, 75, 2 (2009), 109–119.
20. Dohn, N.B. Students' interest in Scratch coding in lower secondary mathematics. *British Journal of Educational Technology*, 51, 1 (2020), 71–83.
21. Drury, B.J., Siy, J.O. and Cheryan, S. When do female role models benefit women? The importance of differentiating recruitment from retention in STEM. *Psychological Inquiry*, 22, 4 (2011), 265–269.
22. Eccles, J.S. Understanding women's educational and occupational choices: Applying the Eccles et al. Model of Achievement-Related Choices. *Psychology of Women Quarterly*, 18, (1994), 585–609.
23. Eccles, J.S. and Wigfield, A. Motivational beliefs, values, and goals. *Annual Review of Psychology*, 53, (2002), 109–132.
24. Fennema, E. and Sherman, J. A. Fennema-Sherman mathematics attitudes scales: Instruments designed to measure attitudes toward the learning of mathematics by females and males. *Journal for Research in Mathematics Education*, 7, 5 (1976), 324–326.
25. Friend, M. Middle school girls' envisioned future in computing. *Computer Science Education*, 25, 2 (2015), 152–173.
26. Google and Gallup Inc. *Diversity gaps in computer science: Exploring the underrepresentation of girls, Blacks and Hispanics*. (Google, 2016).
27. Güler, D. and Camp, T. *Investigating the Incredible Shrinking Pipeline for Women in Computer Science*. (National Science Foundation, 2001).
28. Hall, C., Dickerson, J., Batts, D., Kauffmann, P. and Bosse, M. Are we missing opportunities to encourage interest in STEM fields? *Journal of Technology Education*, 23, 1 (2011), 33–46.
29. Hidi, S. and Renninger, K.A. The four-phase model of interest development. *Educational Psychologist*, 41, 2 (2006), 111–127.
30. Hoyt, C.L. and Simon, S. Female leaders: Injurious or inspiring role models for women? *Psychology of Women Quarterly*, 35, 1 (2011), 143–157.
31. Kafai, Y.B., Fields, D.A. and Searle, K.A. Electronic textiles as disruptive designs in schools: Supporting and challenging maker activities for learning. *Harvard Educational Review*, 84, 4 (2014), 532–556.
32. Kafai, Y., Griffin, J., Burke, Q., Slattery, M., Fields, D., Powell, R., Grab, M., Davidson,

- S. and Sun, J. A cascading mentoring pedagogy in a CS service learning course to broaden participation and perceptions. in *Proceeding of the 44th ACM Technical Symposium on Computer Science Education*. (Denver, CO, 2013), 101-106.
33. Lent, R. W., Brown, S. D. and Hackett, G. Toward a unifying social cognitive theory of career and academic interest, choice, and performance. *Journal of Vocational Behavior*, 45 (1994), 79-122.
34. Lent, R. W., Lopez, A. M., Lopez, F. G. and Sheu, H. B. Social cognitive career theory and the prediction of interests and choice goals in the computing disciplines. *Journal of Vocational Behavior*, 73, 1 (2008), 52-62.
35. Lloyd, A., Gore, J., Holmes, K., Smith, M. and Fray, L. Parental influences on those seeking a career in STEM: The primacy of gender. *International Journal of Gender, Science and Technology*, 10, 2 (2018), 308-328.
36. Margolis, J., Fisher, A. and Miller, F. The anatomy of interest: Women in undergraduate computer science. *Women's Studies Quarterly*, 28, 1/2 (2000), 104-127.
37. Master, A., Cheryan, S. and Meltzoff, A.N. Computing whether she belongs: Stereotypes undermine girls' interest and sense of belonging in computer science. *Journal of Educational Psychology*, 108, 3 (2016).
38. National Science Board. *Science and engineering indicators*. (National Science Board, 2018).
39. Perdikuri, K. Students' experiences from the use of MIT app inventor in classroom. in *Proceedings of the 18th Panhellenic Conference on Informatics*. (Athens, Greece, 2014), 1-6.
40. Pietri, E.S., Johnson, I.R., Majid, S. and Chu, C. Seeing what's possible: Videos are more effective than written portrayals for enhancing the relatability of scientists and promoting Black female students' interest in STEM. *Sex Roles: A Journal of Research*, (2020). Advance Online Publication. DOI:<https://doi.org/10.1007/s11199-020-01153-x>.
41. Rosson, M.B., Carroll, J.M. and Sinha, H. Orientation of undergraduates toward careers in the computer and information sciences: Gender, self-efficacy and social support. *ACM Transactions on Computing Education*, 11, 3 (2011), 14:1-23.
42. Roy, K. 2012. App Inventor for Android: Report from a summer camp. in *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*. (Raleigh, NC, 2012), 283-288.
43. Schunk, D.H. Self-efficacy and achievement behaviors. *Educational Psychology Review*, 1, (1989), 173-208.
44. St-Jean, E., Radu-Lefebvre, M. and Mathieu, C. Can less be more? Mentoring functions, learning goal orientation, and novice entrepreneurs' self-efficacy. *International Journal of Entrepreneurial Behaviour and Research*, 4, 1 (2018), 2-21.
45. Sun, C. *Power of near-peers: Conceptualizing and testing a near-peer mentoring model in raising youths' self-efficacy in computer programming*. (Dissertation: Utah State University, 2020).
46. Sun, C. and Clarke-Midura, J. Testing the efficacy of a near-peer mentoring model for recruiting youth into Computer Science. *Mentoring and Tutoring: Partnership in Learning*, (in press).
47. Tashakkori, R., Wilkes, J.T. and Pekarek, E.G. A systemic mentoring model in computer science. in *Proceedings of the 43rd Annual ACM Southeast Regional Conference*. (Kennesaw, GA, 2005), 371-375.
48. Todd, B. *Little scientists: Identity, self-efficacy, and attitudes toward science in a girls' science camp*. (University of Oregon: ProQuest Dissertations Publishing, 2015).
49. U.S. Census Bureau. *U.S. Census Bureau QuickFacts: Utah*. (2021); <https://www.census.gov/quickfacts/UT>. Accessed 2022 April 13.
50. Urness, T. and Manley, E.D. Generating interest in computer science through middle-school Android summer camps. *Journal of Computing Sciences in Colleges*, 28, 5 (2013), 211-217.
51. Wagner, A., Gray, J., Corley, J. and Wolber, D. Using app inventor in a K-12 summer camp. in *Proceeding of the 44th ACM Technical Symposium on Computer Science Education*. (Denver, CO, 2013), 621-626.
52. Wang, J., Hong, H., Ravitz, J. and Ivory, M. Gender differences in factors influencing pursuit of computer science and related fields. in *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*. (Vilnius, Lithuania, 2015), 117-122.

Chongning Sun, Ph.D.

eLearning Design and Services
University Information Technology Services
Indiana University Bloomington
2709 East 10th Street
Bloomington, IN 47408
csi25@iu.edu

Jody Clarke-Midura, Ed.D.

Jody Clarke-Murray, EdD:
Department of Instructional Technology and Learning Sciences
Utah State University
2830 Old Main Hill
Logan, UT 84321
jody.clarke@usu.edu

DOI: 10.1145/3524916

©2022 ACM 2153-2184/22/06 \$15.00

Publish Your Work Open Access With ACM!

ACM offers a variety of Open Access publishing options to ensure that your work is disseminated to the widest possible readership of computer scientists around the world.



**Please visit ACM's website
to learn more about
ACM's innovative approach
to Open Access at:
www.acm.org/openaccess**



**Association for
Computing Machinery**