International Journal of
Gender, Science and Technology
http://genderandset.open.ac.uk

Designing a Culturally Responsive Computing Curriculum For Girls

Kimberley Scott[1], Gregory Aist[2], Xiaolong Zhang[3]

[1]Arizona State University USA, [2]Iowa State University USA, [3]Microsoft, USA

ABSTRACT

Computer science as a discipline faces challenges in recruiting students, particularly young women, who have been underrepresented from the United States' economically stressed areas (e.g. African American, Latina, and Native American) continue to remain in the minority of the computer science population. One of the many explanations for this disparity is culturally irrelevant computer science activities that fail to consider women's intersecting cultural identities or potential for making a social impact through their innovations. We attended to these issues by designing and implementing culturally responsive computer science exercises for a multimedia program, entitled COMPUGIRLS, targeting adolescent girls (ages 13–18) from the United States' under resourced settings. This case study describes COMPUGIRLS' original iteration, the curriculum design, and the lessons learned in embedding a discipline that historically has not considered cultural issues or social justice within a framework that prioritizes these concepts – a culturally responsive framework. In the end, we consider how instructors might adopt and adapt our process and exercises for various underprivileged communities.

KEYWORDS

gender; culturally responsive; technology; minorities

Designing A Culturally Responsive Computing Curriculum For Girls

INTRODUCTION

Computer science as a discipline faces challenges in recruiting students who have been traditionally underrepresented, particularly young women. Women made up 47% of the North American workforce in 2009, yet were only 25% of those in "computer science and mathematical occupations" and 14% of those in "architecture and engineering occupations" (Mikaye, Kost-Smith, Finkelstein, Pollock, Cohen, & Ito, 2010, p. 1234). Moreover, only 3% and 1% of the United States computing workforce are African American and Hispanic respectively, while White women comprise 18% (NCWIT, 2012). This persistent underrepresentation results in costly consequences for women, the global economy, and society (Ashcraft & Blithe, 2010).

Research indicates a multitude of explanations for the disparities. Among them, factors include: differential technology resources available in home and school settings along gender divides (Varma, 2009) and how female computer science undergraduates view computer use, programming, and peer interaction (Stoilescu & Egodawate, 2010). Such analyses provide the much needed data illustrating gender differences in particular contexts. However, gender is only one unit shaping the contours of our lives.

Few studies specific to technology and technical careers examine how raced-, gendered-, and classed-groups' interests may vary along more divides than are commonly explored (Johnson, Phillips, & Stone, 2008). We believe that it is impossible to design attractive instructional material and/or enrichment programs without careful consideration of the target audience's intersecting cultural identities. Particularly for women of color, cultural identities are informed by race, social class, ethnicity, language, and so forth (Collins, 2000; Crenshaw, 1991). These factors also shape the contours of girls' contextualized developing selves, consciousness, and attitudes including their occupational aspirations. Too few enrichment programs consider these elements and lead to culturally irrelevant programs that ultimately depress engagement of these populations into computer science (Scott & White, 2013).

This case study describes our efforts at creating a culturally responsive multimedia program, entitled COMPUGIRLS, for adolescent girls (ages 13–18)[1]. Although our program has been recognized by the highest office in the United States, we have had some difficulties. This article describes COMPUGIRLS's original iteration, the computer science curriculum design, and the lessons learned in embedding a discipline that historically has not considered cultural issues or social justice within a framework that prioritizes these concepts – a culturally responsive framework. We hope that readers will learn from our challenges and include some aspects of our program as we seek to narrow divides.

COMPUGIRLS

Creating culturally relevant programming is one of two effective strategies for engaging underrepresented students in computer science (Denner & Martinez, 2010; Eisenhart & Edwards, 2004; Rodríguez, Baumann & Schwartz, 2011; Scott & White, 2013). A second approach is the use of hands-on, project-based lessons. We combined both methods in developing COMPUGIRLS. Together, the Principal Investigator (Scott), a computer science colleague (Aist), an expert in Indigenous Communities, and a scholar in game design drew upon their practical and theoretical knowledge to map the Social Justice Youth Development Framework (Ginwright, Noguera, & Cammarota, 2006) onto our multimedia endeavor. Our collective work included putting ideas to an interdisciplinary advisory board that contained two high school-aged participants, academics, community leaders, and school district administrators. For us, culturally responsive computing practices became a means of encouraging participants to collectively conceptualize, engage with, and challenge exploitation and disenfranchisement. In a highly interactive, culturally responsive context, three objectives shape COMPUGIRLS: (1) to use multimedia activities as a means of encouraging computational thinking; (2) to enhance girls' technosocial analytical skills using culturally relevant practices; and (3) to provide girls with a dynamic, fun learning environment that nurtures the development of a positive self-concept as well as effective collaboration skills.

Elsewhere, we discuss the extent to which we met the above objectives. For instance, our application of computational thinking is how confident participants feel while interacting with the technology the program affords. Drawn from a quasi-experimental design in which we tested our assumptions about the participants' motivation and the program's influence on their self-perception as future technologists, findings suggest the following: That over the course of their participation in COMPUGIRLS, their technological self-concept and confidence in operating systems use grew significantly compared to those of a control group (Scott, Husman, & Lee, 2011). Further analyses of qualitative data exploring the motivations of COMPUGIRLS to enroll and persist in the program indicate that the girls' initial understanding of technology's instrumentality is relatively high; what changes as they navigate the program is a burgeoning sense of self as a manipulator of the technology as a technosocial agent (Scott & White, 2013). Finally, another quantitative project in which we associated the growth trajectories of academic possible selves and self-regulation between the program participants and comparison group suggested that COMPUGIRLS experienced a significant increase in their self-regulation (Lee, Husman, & Scott, 2011). In part, we conceptualize these results as products of our culturally responsive efforts.

Culturally Responsive Pedagogy (CRP) refers to a non-traditional teaching approach. In this unique context, teachers and the classroom culture recognize and actively include students' cultural aspects as valuable ingredients in the learning process (Gay, 2000, 2002; Howard, 2001, 2003; Ladson-Billings, 2006; Lee, 2007; Paris, 2012; Santamaria, 2009; Villegas & Lucas, 2004). For our purposes, we focus on three key elements of this framework to ground a discussion of culturally responsive computing (CRC) (see for more detail Scott, Sheridan, & Clark, 2014).

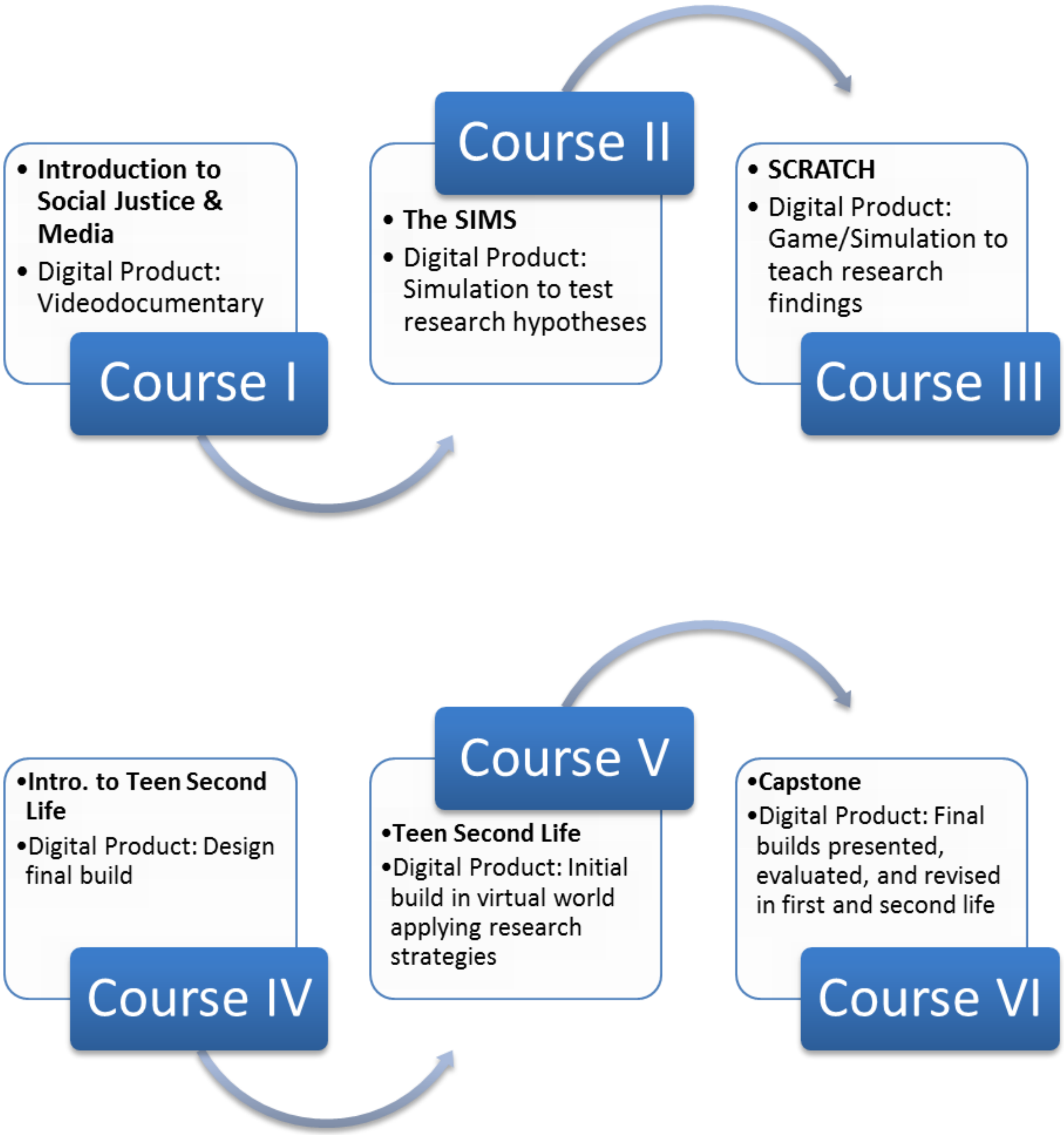
Drawing from CRP's emphasis that students' culturally specific knowledge is valuable and subjective, and should serve as the foundation for new learning (Ladson-Billings, 2006; Lee, 2007), CRC encourages reflection, asset building, and connectivity*.* Specifically, CRC encourages teachers to reflect on their participation in (de)constructing the digital knowledge economy and to develop their technosocial abilities. Asset building in a CRC environment stands in contrast to notions supporting the coding and treatment of communities and social groups as technologically deficient. We define connectivity as a system of interpersonal associations using technology for community advancement.

To include reflection, asset building and connectivity, a typical lesson follows a four-step process. The curriculum requires instructing adults – called mentor teachers (discussed below) – to present either the technical aspects of the media (e.g. how to create a podcast) or social justice concept (e.g. the difference between equity and equality) in a whole-group format. Second, within smaller groups of five, girls have the opportunity to conceptualize, "play with," and present to their peers how they believe the technology or social justice concept can be used to describe and/or analyze their topics (reflection*).* Inevitably, this step requires the girls to learn about the media, its limitations and possibilities, and critique the ways it has been used in the past. The third step involves peers providing presenters with critical feedback to enhance their presentations by responding to the following set of questions: who benefits from the proposed resolution/proposed technology display/research project; who may be disadvantaged; and what are the social and potential individual costs for resolving this issue (asset building). The fourth step for any given lesson involves individual girls assuming roles that may benefit their small group or larger community (connectedness*).* For instance, a daily lesson will again end with a girl presenting to her small group what technical and social justice concepts she has learned today and how she might engage her community in collaborating with her to cause change. Peers elect one girl per group to conduct a whole-class presentation. Fellow COMPUGIRLS again provide feedback to each presenter and verbalize how they see her learning affecting their individual and/or group work. Particular attention is paid to how the effect may cross racial, ethnic, or other socially constructed borders defining difference. As this is by no means a linear process, we encourage girls to provide feedback both in real time and via social networking sites between sessions using our loaned laptops. Since the participants represent various racially and ethnically diverse groups (Latina 74%; African American 19%; Native American 7%), this discussion is rich with insights.

Recruitment

We conduct several recruiting sessions in local schools accommodating economically challenged communities. Targeting rising eighth graders to high school seniors (ages 13–18), these events tend to include staff members showcasing past COMPUGIRLS projects and engaging the potential applicants in an activity they would experience if admitted into the program. Importantly, we encourage all girls in the above age group to apply, even those who have dropped out of school. To this end, recruitment also occurs in non-traditional educational settings (e.g. churches, Boys' and Girls' Clubs).

Figure 1: Course Sequence



Interested individuals submit an application that requests the prospective member to describe why they want to attend the program and what they hope to learn. Three individuals from the program's advisory board evaluate the essays. Their assessments use a ten-point scale with grammar equating to fewer points (3 points) than content (7 points). High-scoring applicants were those who did not necessarily write a tremendous amount or even use proper grammar. Instead, those girls who obtained the higher points articulated their desire to join

COMPUGIRLS in a persuasive way (e.g. I don't have access to technology but know it is important for me to learn if I want to succeed in this world). We also considered technology experience and age by converting the sum of the technology questions into a z-score and adding that to the converted z-score of the personal essay. To further the heterogeneity of the cohorts, we split the summated scores of all applicants into two files according to grade level – namely one file for girls presently in grades 7–8 and the other for girls in grades 9–12. Typically we are oversubscribed, receiving more than 100 applications. To allow for attrition, we admit the top 25 girls from both files in hopes that 40 students will enroll. Once selected, we ask girls to navigate six courses over two years (see Figure 1).

Program Structure
During the summer, participants begin the journey by attending COMPUGIRLS sessions four days a week for approximately 20 hours a week. This intensive period lasts five weeks. Course II and III meet during the academic year typically on one day after school for two hours and for a six-hour Saturday session. Generally, we schedule these sessions for ten weeks, Fall and Spring. We follow this same sequence through Year II to complete Courses IV–VI. Two sites hosted the program – one in a western university classroom with the other housed in a local Indigenous Community's Boys' and Girls' Clubhouse. Although the total number of hours equates to 500, girls often request additional meeting time with mentor teachers to complete their projects.

Mentor Teachers
Trained and compensated mentor teachers become the responsible individuals guiding the girls through all six curricula. Graduate students and in-service teachers from a variety of fields and representing a relatively racially diverse group (e.g. Latino, White, Native American men; White and Latina women) constitute the responsible team of adults. For the graduate students, few have experience working with this age group; however, we specifically target in-service teachers. This allows the graduate students, many of whom are more technologically savvy than the in-service teachers, to learn some classroom management tricks while in-service teachers can benefit from graduate students' technical acuity. Once mentor teachers have completed a fifteen-week graduate level course that offers training over 45 hours of coursework, we assign a group of five girls to at least one adult. These individuals guide girls to create both a digital product using loaned hardware (cameras, camcorders, laptops) containing necessary programs and a research paper.

Closing ceremonies signal the end of each course. During these community events, girls organize a program to present their projects to families, peers, and community leaders. The hope is that their demonstrations will encourage others to assist them in doing something meaningful in light of their research and suggestions.

CURRICULUM DEVELOPMENT
Three recent developments in research on the pedagogy of computer science as it relates to race and gender are salient here. The first is recent work on approaching computer science using problem-solving strategies, rather than a solutions-based

approach. This requires a focus on computational tools and techniques in a course-by-course approach (Fee & Holland-Minkley, 2010). The second is a method that uses service learning to attract diverse students, develop leadership skills, and foster productive student attitudes towards the discipline. The third is the increasing discussion about expanding the content of computer science education to include societal impact, interaction skills for group work, and the application of computing (Stoilescu & Egodawate, 2010; Koppi, Sheard, Naghady, Edwards, & Brookes, 2010). In the COMPUGIRLS project we draw from a problem-focused approach.

The Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics Engineers (IEEE) Computer Society have collaborated to produce a curriculum for undergraduate-level study in computer science and related disciplines. The combined standards from the two professional societies present a broader view of information and computing technology than the Computer Science Teachers' Association (CSTA) K-12 standards. We drew 14 main focus areas from their curriculum. From each of those foci, we chose one instructional point to convey during the social justice and computing exercises. While identifying the specific instructional points we took several factors into account.

First, we wanted the material to lend itself to a brief familiarization lesson. Then we wanted the material to be concrete rather than abstract. Finally, we wanted the material to be visualizable in an intuitive way – not necessarily literally graphical, but at least to have a visual or physical analogue that could be worked with in a group setting. For example, the Discrete Structures (DS) area includes six subareas: Functions, Relations, and Sets; Basic Logic; Proof Techniques; Basics of Counting; Graphs and Trees; and Discrete Probability. Of these six topics, Functions, Relations, and Sets is fairly abstract, as are Basic Logic and Proof Techniques. Basics of Counting, Graphs and Trees, and Discrete Probability are more concrete; Graphs and Trees have a direct visual analogue and Discrete Probability has a direct physical analogue. For each subarea we then identified a specific instructional point for which we could draw a link to an area of concern for social justice: culture, language, ability, geography, race/ethnicity, and gender. During this selection process, three COMPUGIRLS team members (e.g. one faculty member in Computer Science and two graduate students in Computer Science) worked with three to four topic areas to independently propose a selection of subarea choices along with corresponding instructional points. This panel then discussed the proposed focus areas and instructional points in order to arrive at a consensus for what would be concrete, visualizable, and accessible to the target population. We reviewed these choices with the other COMPUGIRLS principal investigators and graduate students in education to refine our proposal.

Several brainstorming sessions were held to generate ideas on how to convey these instructional points in a culturally responsive way. Ultimately, our discussions caused us to outline a lesson for each point. Re-analysis of the sociocultural content of the lesson outlined led to the identification of six general areas of culturally specific interest: (1) culture, including material culture, social aspects, special observances, and so forth; (2) language, including language heritage, writing systems, and language learning issues; (3) ability, including both ability and disability across several dimensions such as sight, hearing, and mobility; (4) geography, including urban/rural distinctions, terrain, and climate; (5) race/ethnicity, including a wide variety of issues relating to race and/or ethnic background; and (6) gender, including both opportunity and discrimination and identity.

During the development process, which we describe next, specific issues within these general areas were used to illustrate, motivate, and serve as a test bed for the 14 instructional points from computer science.

We developed a pilot set of lessons based on a selection from the 14 points and the six issue areas. While establishing the lessons we maintained several principles:

- The structure of the social justice issue should be analogous to the structure of the computer science representation.
- The social justice issue should be related to the practical aspects of the COMPUGIRLS' lives.
- The computer science exercise should produce some product that is written down on paper or electronically, rather than an understanding that is tested later.
- The exercise and product should be something that can be personalized in some way, or enable ownership by the girls. For example, if the exercise is "a day in the life of a text message," the girls could use their own phone and network as the example, rather than a single fixed example.
- The mathematics should be able to be represented visually – for example, what numbers a floating point number can represent could be illustrated with a number line; how a quadtree encodes an image could be represented with a picture with boxes drawn on it.

Five pilot lessons were developed. The first was on Information Management (IM) with the topic of Geographic Information Systems (GIS) and how food availability varies from neighborhood to neighborhood. The second was for Software Engineering (SE) on the topic of software requirements and testing, how that relates to computerized voting, and what computer technology is used with particular racial communities to electronically collect and/or count ballots in elections. The third was on Graphics and Visual Computing (GC) on the topic of avatars and ethnic identity. The fourth was on Algorithms and Complexity (AL) on the topic of best-/average-/worst-case analysis of public transportation by neighborhood. The fifth was on Programming Fundamentals (PF) on the topic of conditional and iterative control structures and breaking the cycle of violence.
The initial lessons contained an outline with multiple steps:

1. Introduce the topic
2. Introduction key underlying issues
3. Introduce the social justice challenge
4. Class discussion of how the social justice question affects the participants
5. Describe the data involved in the social justice issue

6. Describe the processes involved in the social justice issue
7. Introduce a computational method of analyzing the issue
8. Present the general computer science question
9. Group work to apply the computer science method to the social justice question
10. Describe one possible solution to the social justice challenge

Several of these lessons were piloted during COMPUGIRLS summer sessions. The pilot lessons were performed in a combination of whole class lectures and small group work breakouts. Each session took approximately one hour. The presenters' background included an undergraduate degree in computer science or a related field, together with completion (for faculty) or enrollment (for graduate students) in a Ph.D. program in computer science or a related field. The presenters were thus fluent in the technical material being presented. In addition, each of the presenters also had some experience of working with K-12 students in a classroom context, either from having worked with COMPUGIRLS or from previous research project experience. Drawing on the presenters' observations, the results of the pilot study are discussed in the next section.

LESSONS LEARNED

You will recall from the previous section that the initial lessons contained ten steps interweaving discussion and group work, and both social justice content and computer science content. When these lessons were piloted during COMPUGIRLS summer sessions, participation was generally good during the class discussion and group work portions. As the COMPUGIRLS worked through the lessons, there was evidence of them adopting the language and methods of the computer science approaches in order to address the matters at hand. For example, in the programming fundamentals session, as the students discussed the life choices they were concerned with, they put together flowcharts illustrating those choices and the subsequent options available. These flowcharts were not just a single loop or a single branch but were also combinations of several loops or branches in order to illustrate the impact of one choice on subsequent choices. While these accomplishments were helpful, we learned equally important lessons from the challenges. We identified three lessons from implementing the pilot exercises:

Lesson #1: The format of the lesson is vital. The original ten-point outline contained too much lecturing for this particular combination of audience and subject matter. However, if all activities were done in small groups, a group with an unhealthy dynamic would not have anything to counterbalance it. In addition, since the lessons generally open with something relevant to the girls but also novel to them, presenting this motivating example in a whole-group setting provides everyone with the same context for group activity and for later discussion.

Lesson #2: Careful selection of the specific representation of the concept, data, or algorithm is essential. The idea needs to be both culturally relevant to the participants and also illustrative without loss of generality. For example, in the lesson on avatars, the motivating concept was of an online (Second Life) hair shop called Dernier Cri which was among the first to allow selection of a full range of hair colors. Dernier Cri was founded by a teen on the teen grid and then moved to the main grid. This example was not only computationally informative (color space) but also culturally relevant, and furthermore provided the COMPUGIRLS' participants with an aspirational peer role model – someone that they could envision becoming like as they entered their careers.

Lesson #3: It is important to use the minimal example necessary to illustrate the underlying idea. An example with too much extraneous detail might lead to the discussion veering off topic. For us, the initial introduction of the Algorithms and Complexity (AL) topic included a discussion of the various kinds of transportation, both public and private, but students began cataloging all of the different ways that they got to and from school and activities. In this exercise, the adjustment to make would be to focus on just one kind of public transportation and to work directly from a map or GIS – for example, an online map of the bus or subway system in the students' city.

CONCLUSION

We aimed to create in our multimedia enrichment program – COMPUGIRLS – a rich computational environment that included two significant elements: (1) features from the real world for girls to analyze and (2) manipulation and customization. In conjunction with the National Educational Standards' description of students' technology skills, we sought to create computer science lessons that allowed students to "use technology to learn rather than learning to use technology" (Denner & Martinez, 2010, p.206). In such a context, computer science activities should develop students' computational problem-solving abilities by applying relevant pedagogical strategies (Fee & Holland-Minkley, 2010). The problem-solving tasks are carefully tied to real-world practice that is relevant to the girls' specific cultural background. Stated differently, COMPUGIRLS aimed to connect computational lessons to the girls' understanding, seeing that, as DiSalvo and Bruckman (2011) found with their game testing program for adolescent African American boys, "Connecting to the deep concept of values is much more important than the surface notion of interests" (p. 29).

Our computer science goal was to develop a "sampler" with the aim of presenting a range of topics that might interest girls in computing. It is thus a more informal and less focused approach than might be appropriate in a for-credit university course. For example, a university course might focus on computers and society, and include fewer of the programming-oriented concepts such as digital representation and kinds of programming languages. In addition, while the curriculum areas and topics are drawn from the ACM/IEEE curriculum, the learning goals were designed to be suitable for the informal setting, the age and background knowledge of the participants, and the overall goals of the COMPUGIRLS program. Those learning goals were not the same as those in the original ACM/IEEE curriculum. A university course might have objectives that are closer to those in the ACM/IEEE curriculum. Equally important, some may say that the girls' enthusiasm is a result of our selection process – that is, they were chosen to participate in a "special" opportunity. Perhaps our findings would differ in an obligatory setting. Also, we did not account for the influence of peer networks and family support on the girls' technological interest. These and other concepts shape our future plans, including how the specific elements of the program determine their influence on the girls' identities. Notwithstanding these limitations, we did document the effects of the COMPUGIRLS experience on the participants and they were generally positive (see for example Scott, Husman, & Lee, 2011; Lee, Husman, & Scott, 2011; Scott & White, 2013).

For this article we have looked at designing culturally responsive computing exercises for young women of color. Readers, no doubt, have their own constituencies to serve, who may or may not be drawn from the same demographics as COMPUGIRLS. Thus it is worth conducting a thought experiment aimed at discovering how the instructional design principles laid out in this article can be extended to other situations and populations: understanding students, for example, who have been in the workplace and are likely to have experienced hourly scheduling (deciding who gets what "hours" to work in a restaurant or retail store) or weekly scheduling (asking for particular days or holidays off). This information is a perfect introduction to two aspects of computing: resource allocation (matching workers to hours), and computer-mediated communication in the informatics sense (using a database system to handle/manage requests for hours). Use of such a scenario in a classroom of students who mostly never had to request a holiday – as might be the case in a room full of first-year undergraduates from privileged backgrounds – would not make sense, yet in a roomful of seasoned career changers this might well be a natural transition to thinking about the problem in a computational way.

In general, the kinds of culturally specific knowledge vary based on the background of the students. Yet the development process is substantially the same, although we recommend neither "top-down" nor "bottom-up" approaches. A "top-down" approach tends to begin with refining teaching and learning goals and then each sub-goal is explained in its own terms; a "bottom-up" instructional approach often focuses on how student interests alone drive the content of the material. Our findings suggest that a culturally responsive approach may be better suited particularly for underrepresented populations. Starting with both a culturally relevant theme and an instructional goal, thus preserving the potential to introduce students to specific topics that take them out of their comfort zone and introduce them to computational topics, may direct their steps down an unanticipated path.

ENDNOTES

1. Funding for this project was made possible through the National Science Foundation Grant #0833773

REFERENCES

Ashcraft, C., & Blithe, S. (2010). *Women in IT: The facts*. Boulder, CO: National Center for Women & IT.

Collins, P. H. (2000). *Black Feminist Thought*. 2nd ed. New York, NY: Routledge.

Crenshaw, K. (1991). Mapping the margins: Intersectionality, identity politics and violence against women of color. *Stanford Law Review 43(6)*, 1241–1299.

Denner, J., & Martinez, J. (2010). Whyville versus MySpace: How girls negotiate identities online. In S. R. Mazzarella (Ed.), Girl Wide Web 2.0: Revisiting Girls, the Internet, and the Negotiation of Identity (pp. 203–222). New York, NY: Peter Lang Publishers.

DiSalvo, B., & Bruckman, A. (2011). "From Interests to Values: Computer science is not that difficult but wanting to learn it is." Communications of the ACM, August 2011, pp. 27–29.

Eisenhart, M., & Edwards, L. (2004). Red-eared sliders and neighborhood dogs: Creating third spaces to support ethnic girls' interests in technological and scientific expertise. *Children, Youth and Environments, 14(2)*, 157–177.

Fee, S. B., & Holland-Minkley, A. M. (2010). Teaching computer science through problems, not solutions. *Computer Science Education*, *20(2)*, 129–144.

Gay, G. (2002). Culturally responsive teaching in special education for ethnically diverse students: Setting the stage. *International Journal of Qualitative Studies in Education*, *15(6)*, 613–629.

Gay, G. (2000). Culturally responsive teaching: Theory, practice, & research. New York, NY: Teachers College Press.

Ginwright, S. A., Noguera, P., & Cammarota, J. (Eds.). (2006). *Beyond resistance!: Youth activism and community change: New democratic possibilities for practice and policy for America's youth*. New York, NY: Routledge.

Howard, T. C. (2003). Culturally relevant pedagogy: Ingredients for critical teacher reflection. *Theory into Practice*, *42(3)*, 195–202.

Howard, T. C. (2001). Telling their side of the story: African-American students' perceptions of culturally relevant teaching. *The Urban Review*, *33(2)*, 131–149.

Johnson, R. D., Stone, D. L., & Phillips, T. N. (2008). Relations among ethnicity, gender, beliefs, attitudes, and intention to pursue a career in information technology. *Journal of Applied Social Psychology*, *38(4)*, 999–1022.

Koppi, T., Sheard, J., Naghady, F., Edwards, S. L., & Brookes, W. (2010). Towards a gender inclusive information and communications technology curriculum: A perspective from graduates in the workforce. *Computer Science Education, 20*, 265–282.

Ladson-Billings, G. (2006). Yes, but how do we do it? Practicing culturally relevant pedagogy. *White teachers/diverse classrooms: A guide to building inclusive schools, promoting high expectations, and eliminating racism*, 29–42.

Ladson-Billings, G. (2004). New directions in multicultural education: Complexities, boundaries, and critical race theory. In J. A. Banks & C. A. M. Banks (Eds.), *Handbook of research on multicultural education.* 2nd ed. (pp. 50–65). San Francisco, CA: Jossey-Bass.

Lee, C. (2007). *Culture, literacy, & learning: Taking bloom in the midst of the whirlwind.* New York, NY: Teachers College Press.

Lee, J. E., Husman, J., & Scott, K. A. (2011, November). *COMPUGIRLS: Stepping stone to the future STEM Pathway.* Poster session presented at the biannual meeting of the Southwest Consortium for Innovative Psychology in Education, Norman, OK.

Miyake, A., Kost-Smith, L. E., Finkelstein, N. D., Pollock, S. J., Cohen, G. L., & Ito, T. A. (2010). Reducing the gender achievement gap in college science: A classroom study of values affirmation. *Science, 330(6008),* 1234–1237.

National Center for Women and Information Technology (2012). *By the numbers*. Retrieved February 15, 2013 from http://www.ncwit.org/resources/numbers

Paris, D. (2012). Culturally sustaining pedagogy: A needed change in stance, terminology, and practice. *Educational Researcher*, *41(3)*, 93–97.

Rodríguez, M. M. D., Baumann, A. A., & Schwartz, A. L. (2011). Cultural adaptation of an evidence based intervention: From theory to practice in a Latino/a community context. *American Journal of Community Psychology*, *47(1–2)*, 170–186.

Santamaria, L. J. (2009). Culturally responsive differentiated instruction: Narrowing gaps between best pedagogical practices benefiting all learners. *The Teachers College Record*, *111(1)*, 214–247.

Scott, K. A., Sheridan, K., & Clark, K. (2014). Culturally responsive computing: A theory revisited. *Learning, Media & Technology.*
DOI: 0.1080/17439884.2014.924966

Scott, K. A., & White, M. (2013). COMPUGIRLS' Standpoint: Culturally responsive computing and its effect on girls of color. *Urban Education, 48–5,* 657–681.

Scott, K. A., Husman, J., & Lee, J. (2011, September). *Motivation and culturally responsive technology for COMPUGIRLS.* Itest Youth Motivation Convening, Boston, MA.

Stoilescu, D. & Egodawate, G. (2010). Gender differences in the use of computers, programming, and peer interactions in computer science classrooms. *Computer Science Education, 20*, 283–300.

Varma, R. (2009). Gender differences in factors influencing students towards computing. *Computer Science Education*, *19(1),* 37–49.

Villegas, A. M., & Lucas. F. (2004). Diversifying the teacher workforce: A retrospective and prospective analysis. *Yearbook of the National Society for the Study of Education, 103(1)*, 70–104.