DOI:10.1145/1978542.1978552

Betsy DiSalvo and Amy Bruckman

Education
From Interests to Values

Computer science is not that difficult but wanting to learn it is.

Computer science is not that hard. Sure it takes work and perseverance, just like learning a second language or studying political science or a hundred other things. But the media depicts computer science as a gift that some people are born with. Movies would lead us to believe that a magical computer gene enabled Mark Zuckerberg to start coding as a teen with the only major side effects an undying thirst for Mountain Dew and social awkwardness. But computing isn't magic fairy dust in our genes. People become computer scientists because they put in the work and have good strategies for how to learn.

We can teach the strategies about how to learn computer science, but how do we get the students to sign up for a computer science class when they are afraid they don't have some innate talent or that CS is just too hard to bother with? For many groups, this may be part of what is keeping them from participating in equal numbers in CS. The National Science Foundation (NSF) has tried to reach these groups, to broaden participation in computing by increasing the number of women, people with disabilities, and Latino, Native Americans, and African Americans. We should continue these efforts, because without these groups participating fully in CS we are recruiting from only 35% of the U.S. population. We will not be able to fill jobs or compete in the marketplace without approaches that motivate young people from a broad set of demographics to persevere in CS. But more importantly the underlying design of computational devices will be based upon the values of a few rather than multiple perspectives that will increase innovation and appropriateness of our designs for the audiences they serve.

Glitch game testers evaluate a pre-release version of NCAA Football by Electronic Arts.

PHOTOGRAPH BY ROB FELT

What does it take to get these groups interested in and then motivated enough to put in the hard work to succeed in computer science? Beyond what they are interested in, what do they *value*? By this we mean in particular, how do they see their current and future role in the world, and how does that shape what they find compelling and worthwhile? Many interventions aimed to broaden participation in computing begin with what underrepresented youth find "cool" or "fun." However, most teenagers—especially those from economically less advantaged backgrounds—have issues more pressing than fun on their minds. They worry about their economic futures in particular. Helping them to imagine new possible futures for themselves is a daunting task. We need to stop looking at what motivates computer scientists and start looking at what motivates our target audience. This may mean segmenting the audience and listening to their needs as groups based on the intersection of gender, race, geography, and economic status. It definitely means that one size does not fit all. A program we have developed, the Glitch Game Testers, is showing promise in this approach by focusing on working with young African-American males to design a CS education program.

The Glitch Project

Glitch is a project of Amy Bruckman's Electronic Learning Communities

Lab and the dissertation work of Betsy DiSalvo in the School of Interactive Computing at Georgia Tech. Glitch is a game testing work program that started in 2009. It is made up of African-American male high school students, with a total of 25 participants over the last two years and little attrition except for those who have graduated. Glitch operates out of a lab in the College of Computing Building at Georgia Tech in partnership with Morehouse College.

Testers are paid to work full-time in the summer and part-time during the school year testing pre-release games for companies like Yahoo!, Cartoon Network, Last Legion, and Kaneva. The program was developed with a demonstration grant from the NSF's Broadening Participation in Computing program under the direction of Jan Cuny. But corporate support has been available from the beginning with Electronic Arts Tiburon helping shape the program and providing Quality Assurance training to the staff and participants, and more recently with support from Yahoo! and Microsoft.

These prestigious companies are working with Glitch, but the testers come from a different world. The majority of testers qualified for free or reduced lunches, attend poorly ranked schools, and are generally not at the top of their class. They were recruited by word of mouth, which resulted in many more applicants than the program could handle. So they were primarily accepted on a first-come, first-served basis. Even though they are coming from rough neighborhoods, these high school students are interacting with developers from game companies they admire. They work just like professional game testers with deadlines and expectations placed on them. The testers are paid hourly, and spend one hour a day on CS class work taught by African-American male undergraduate students from Morehouse and Georgia Tech.

In the first year of class, the testers worked in a computational media approach,[4] integrating the drag-and-drop programming environment Alice with Python programming. The testers who were entering their second year in the program wanted to do something more challenging, something more like "real programmers." They were offered an Advanced Placement Computer Science (APCS) class that teaches deeper CS concepts with Java—and all students elected to take it.

Finally, both first and second year students build computers. This serves two purposes beyond learning about computer hardware. First, they needed higher-quality machines to test on—the program was started using surplus machines from Georgia Tech and Morehouse and building them was cheaper than buying them. Second, one of the computers is the final prize in the Glitch "point competitions." Each school year and each summer the participant who earns the most points from bug reports, regression testing, CS assignments, and other tasks earns a computer. This competitive environment plays an important part in motivating students, not only to work hard to earn points but also as a "face saving" excuse when challenged about why they are putting in extra effort. In teen culture, it is sometimes perceived as "not cool" to try hard. But trying to win a computer is a socially acceptable reason to work hard—and our students do.

A number of measures including computer science tests and self-assessments indicate Glitch participants have increased their knowledge about software development and programming. Pre- and post-surveys indicate that Glitch has changed attitudes about studying computing and intent to persist in computing. For example, 22 out of 23 testers last summer reported an increase in their interest in technology. Perhaps the most compelling indication of success is what the students are doing when they leave. Out of the seven students who graduated from high school in 2010, six are attending college—five of them in computing related majors, (two in CS, one in computer engineering, two in digital media).[3] Of the seven students who will be graduating in 2011, all intend to attend college—four have declared their major as CS and the other three are considering CS as their major. Before the program started, of the 14 only one had an interest in CS and one in computer engineering as a major. These are amazing numbers for any type of computer science outreach, but particularly with a group that is usually so difficult to reach.

Being a computer programmer was no longer a mystical quality that was out of their reach...

Why did it work, and how can others do similar things? The Glitch program motivated the students to put in the work to do computing. Being a computer programmer was no longer a mystical quality that was out of their reach but a natural progression from what they were already doing as game testers. They communicated with developers daily, and felt that developers really listened to them, changing games based upon what they had written in bug reports. They already felt like a valuable part of the software development process, so imagining themselves as a bigger part of that process was easy.

Challenges

There are still significant hurdles. From our staff's assessments, four of the seven students who participated in the Glitch APCS classes were prepared to pass the test in May of this year, but only one chose to take the test. We are trying to understand why these young men who had worked so hard all year learning introductory CS would not be interested in sitting for the test. They claim that logistics was part of the reason; the test was offered at 8:00 A.M. at a different school than they attended, on a school day, and none of them wanted to wake up that early to take public transportation there and miss school. But we suspect another factor was intimidation. Perhaps they feared taking the test at school that might be perceived as "better," or not doing well and losing the face of being tech savvy to their family and perhaps themselves. We have started to ask them questions and would like to gain a better understanding of why they did not

take the test. Like all teens, they are not always consciously aware of their own motivations. The challenge is not just to encourage them to share their reasons, but to help them develop greater self-awareness and the ability to articulate their conscious and subconscious motivations.

And these questions may have larger implications for African-American teens across Georgia. The Georgia Computes Broadening Participation in Computing Alliance[1] has made increases in the number of APCS test takers (from 389 when the program started in 2004 to 692 in 2010), However, the percentage African-American test takers has remained flat. In 2010, only 9.8% of the test takers were African American compared with 29% of the Georgia population. Even more shocking is how few African-American students pass the APCS exam. Only 16 of 68 (23.5%) African-American students passed in 2010, compared to an overall pass rate of 56%.[2] Connecting the value of taking the APCS test to the values of our African-American students will be important for our future efforts.

Games are an obvious interest match for young African-American men—they play often and games are reported to be the digital media they interact with most frequently. Creating a competitive environment went along well with their values on competitive gaming practices and the quota-based game testing industry. Finally, these young men needed jobs. They could not justify traveling to Georgia Tech every day, or even every week to just learn about computers. The pragmatic approach they take to their extracurricular activities needs to be reflected in the values of the program.

The challenge is to identify interest and values in other groups and design educational programs around them. While a number of programs use games, robots, or other "fun" computing tools to get kids interested, maybe we need to step back and stop trying to make computing look cool and start looking at audiences' values. While these young African-American men were attracted to game testing because it was cool, they were able to participate because it was a paying job that met needs they valued more than an interesting thing to do with their time. They want to continue in computer science because they now see CS as a pragmatic approach of an attainable high-paying career, something they and their families value.

Conclusion

We began to understand the values of young African-American men from underserved neighborhoods through in-depth formative work consisting of observations, participatory design activities, interviews, focus groups and prototyping. And after two years of running the Glitch program, we can begin to speak with authority of what these values are and how they impact learning CS. Many interventions designed to make computing engaging for students focus on their interests. It definitely helps to take elements from popular culture and the arts that appeal to your target audience and leverage those interests. But fewer interventions leverage students' *values*. What is "fun" and what is "important" for kids are two different things. For teens that grow up in poverty, finding a path to economic security is critical. And it has to be an accessible path—one they can easily imagine themselves on. Values connect to what they view as important, who they see themselves as now, and who they can imagine becoming. The observation that African-American teenage boys often like video games was a first step for our design of Glitch, but perhaps the least important one. Connecting to the deep concept of values is much more important than the surface notion of interests. C

References

1. Bruckman, A. et al. *Georgia Computes!: Improving the Computing Education Pipeline.* Paper presented at the SIGCSE 40th ACM Technical Symposium on Computer Science Education Chattanooga, TN, 2009.
2. CollegeBoard. *Advanced Placement Testing National Summary Reports: 2010,* (2010).
3. DiSalvo, B. et al. *African American Males Constructing Computing Identity.* Paper presented at the ACM CHI Conference on Human Factors in Computing Systems, Vancouver, BC, 2009.
4. Guzdial, M. A media computation course for non-majors. *SIGCSE Bulletin 35,* 3 (Mar. 2003), 104–108.

Betsy DiSalvo (bdisalvo@cc.gatech.edu) is a Ph.D. student in the School of Interactive Computing at Georgia Institute of Technology.

Amy Bruckman (asb@cc.gatech.edu) is an associate professor in the School of Interactive Computing at Georgia Institute of Technology.