Let's All Learn to Code

The Importance of the Involvement of Underrepresented Groups in Learning Coding and Computer Science

Tisha Savannah December 10, 2019 TCOM 5340

Introduction

The fields of STEM, which stands for Science, Technology, Engineering, and Mathematics, are more recently being focused on as industries are in need of employment of knowledgeable people in these industries, more specifically, underrepresented groups of women/girls and people of color. Learning to code, or computer programming, is a part of STEM that is heard about more and more on a daily basis. Why do people code? People code for various reasons: some want to make a program that will help them in their daily routine, others code programs as a form of self-expression. "These reasons support both the intended expressive end uses of the programs and the readability of their code for human (developer) audiences" (Brock and Mehlenbacher, p. 388). In his TED Talk, Code.org co-founder Hadi Partovi makes the point that coding is a form of science: computer science. He made a statement that technology touches everything, meaning that technology is not only restricted to one field, but it is a driving factor in many fields, from architecture to the medical field and

beyond. Partovi also discussed that there are still many schools that do not teach computer science, and asked that if computer science is not taught in all schools, how are students supposed to be ready to take on jobs that require this knowledge if they are not being educated in this field? In considering the implementation of additional computer science courses across the country's schools, this leads me to ask what are the plans in place to increase the number of computer science courses available? What can be done to increase interest and make learning coding more easily accessible to these underrepresented groups? How can technical communication help in these efforts? In this document, I will explore the processes involved in learning and teaching to code, how the aforementioned underrepresented groups are affected in these processes, while evaluating the role technical communication plays in those processes.

Learning Code is Important

In Ashley Gavin's TED talk, she brings up the point that Computer Science must be applied to other fields, it cannot exist alone. We find this to be true because a computer science project must be initiated by the need of another field. You can code a program, but what that program needs to achieve will be outside the realm of computer science.

Learning to code is learning a branch of computer science. As Lussos discusses in her article, when her students learned to code Twitter bots, there are a number of steps the student must reason through in the process of programming a Twitter Bot: "…to compose a Twitter bot, students must first conduct rhetorical analyses of purpose, audience, and delivery before they can begin "writing" (i.e., programming) the final product" (Lussos, p. 1). The thought processes learned in this type of exercise demonstrates skills students will carry with them to apply in other areas of learning. The processes involved with game design are similar, while also incorporating elements of technical communication: "Computer game design is a field of technical communication not because it relies on computer code as a central technology (although it sometimes can), but because the fundamental processes and practices of game design are technical practices such as iterative design, testing, and even the production of algorithmic rule systems that structure player experience" (DeAnda and Kocurek, p. 203). Also of note, the composition of the lessons that are being used to teach coding is also an element in which technical communication is extremely important.

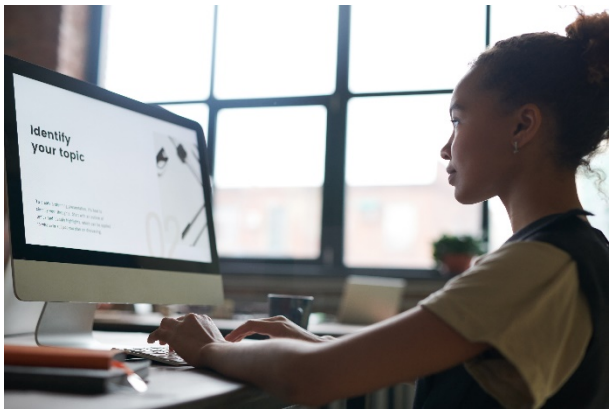
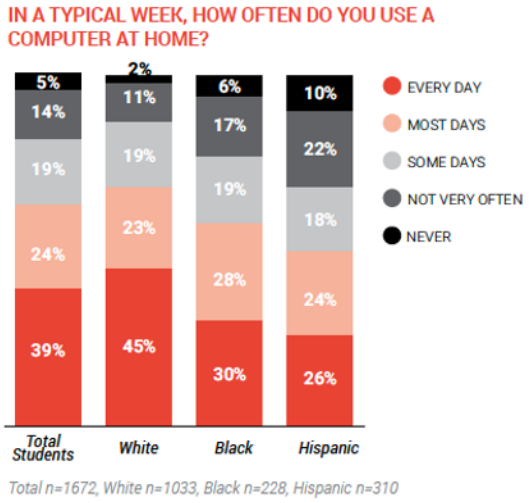


Photo by fauxels (pexels.com).

Student Exposure to Computer Science

While online learning is extremely helpful, the opportunity to access a physical course to attend is important, as there are instances in which students may not have access to a computer they can use at home. A study performed by Google and Gallup (2016) gives the following powerful statistic regarding at-home computer usage:

> Overall, almost four in 10 students (39%) use a computer at home every day, and among those who do, more than three-quarters (77%) use a computer at home for two hours or more daily. White students are more likely than Black and Hispanic students to use a computer at least most days of the week at home. In fact, two-thirds of White students (68%) use a computer at home at least most days a week, while just half of Hispanic students (50%) use a computer that often. Almost six in 10 Black students (58%) use computers at home at least most days of the week. Just one in 20 students (5%) say they never use a computer at home. (p. 12)



Google and Gallop, 2016.

While to most people, at-home computer usage is a given, it should not be assumed that computers are easily accessible to everyone.
Access to technology is only a portion of the problem. Another issue is to not only get underrepresented groups involved in Computer science, but to encourage them that it is indeed something they can excel in doing.

An ongoing issue is females are apprehensive to learn computer science. “Girls don’t want to be in courses if they don’t feel a sense of belonging,” (Code.org, 2019). An additional factor is the attitudes of parents and teachers towards girls learning computer science. “If [parents, teachers and principals] believe that certain students are not interested, then they may be less likely to encourage or even expect those students to learn CS” (Google and Gallop, p. 25). In focusing on the involvement of this underrepresented group, it is possible for attitudes towards computer science. Activities like Hour of Code, or introduction events held by schools can go a long way towards positively impacting attitudes toward computer science and show that it is for everyone.

Playing to Learn

Code.org has a program they call Hour of Code. This encourages young users to take one hour out of their day to learn to code. Hour of Code uses play and familiar characters from popular culture to instruct young users how to code, so it is more like they are playing games.

> Findings show that after completing one Hour of Code activity, students' positive attitudes towards computer science increase along with feelings of computer science self-efficacy, especially for female students. In other words, after just one Hour of Code activity, students report liking computer science more and report feeling that they are better able to learn computer science and are better at computer science than their peers. (Philips and Brooks, p. 1)

Faris, et al., discuss how play was important in their experiment: "We were learning to play with the composition process by using technologies that we may not have normally considered technologies appropriate for composing. This process challenged and broadened our received definition of composing" (Faris, et al., 2018). In making the lessons fun and less like "learning" or "science," they become more attractive to students that would ordinarily have little interest in learning the material.

Conclusion & Recommendations

Technical communication is being used to help instructors in efforts to teach coding and computer science by taking into consideration the means of presenting the information for learning. Students need advocates to encourage them they can succeed in computer science despite it looking somewhat intimidating at first glance. Several websites, one being Code.org, offer curriculum for free for anyone wanting to instruct students to learn to code, thus helping to make this information available to an increasing amount of people. There is still much work to be done in working towards a society in which coding and computer science are in every household, like reading and math. Code.org, and other websites like them such as GirlsWhoCode.com, BlackGirlsCode.com, KhanAcademy.org, and Codecademy.com to name a few are making efforts daily to ensure that we continue to move in the direction of coding and computer science being available for everyone.

Photo by Christina Morillo (pexels.com).

Recommendation 1: Increase exposure to information about computer science. Classroom coding activities or introduction events held by schools can go a long way towards positively impacting attitudes toward computer science and show that it is for everyone.

Recommendation 2: Utilize lesson plans to teach computer science that are inclusive of all groups, including underrepresented groups.

An example of this is how Code.org is using their website to show more women and people of color involved with computer science; they are also gearing their lessons to use gender neutral avatars and intellectual properties ranging a variety of interests, from Disney to DreamWorks to Minecraft, to increase interest in the coding lessons offered.

Recommendation 3: Demonstrate how everyday actions like knitting and stop lights show patterns and recognize that pattern recognition skills correlate with coding and computer science. Address how concepts learned in computer science can be applied to learning a multitude of other subjects.

References

Brock, K. & Mehlenbacher, A.R. (2018). Rhetorical Genres in Code. *Journal of Technical Writing and Communication, 48*(4)*,* 383-411. DOI: 10.1177/0047281617726278

Code.org (2019). 'Hey, I'm good at this!': One school's solution for recruiting more girls to AP CS. Retrieved from https://medium.com/@codeorg/how-to-recruit-more-girls-for-ap-cs-classes-8794a28c324c

DeAnda, M.A. & Kocurek, C.A. (2016). Game Design as Technical Communication: Articulating Game Design Through Textbooks. *Technical Communication Quarterly. 25*(3), 202-210. http://dx.doi.org/10.1080/10572252.2016.1185161.

Faris, M.J., Blick, A.M., Labriola, J.T., Hankey, L., May, J., & Mangum, R.T. (2018). Building Rhetoric One Bit at a Time: A Case of Maker Rhetoric with LittleBits. *Kairos: A Journal of Rhetoric, Technology, and Pedagogy, 22*(2). Retrieved from http://kairos.technorhetoric.net/22.2/praxis/faris-et-al/conclusion.html#

Gavin, A. (TEDx Talks). (2015, July 21) Computer Science Education: Why Does It Suck So Much and What If It Didn't? [Video File]. Retrieved from https://youtu.be/5jmN_tBS0t4

Google Inc. & Gallup Inc. (2016). Diversity Gaps in Computer Science: Exploring the Underrepresentation of Girls, Blacks and Hispanics. Retrieved from http://goo.gl/PG34aH.

Lussos, R. G. (2017). Twitter Bots as Digital Writing Assignments. *PraxisWiki Kairos: A Journal of Rhetoric, Technology, and Pedagogy, 22*(2). Retrieved from http://praxis.technorhetoric.net/tiki-index.php?page=PraxisWiki:_:twitterbots#Benefit_Four:_Twitter_Bots_Introduce_Students_to_Programming_Logic_Without_Learning_a_Language

Partovi, H. (TEDx Talks). (2014, December 10). *Computer Science is Foundational* [Video file]. Retrieved from https://youtu.be/m-U9wzC9xLk

Phillips, P.S. and Brooks, B.P.C. (2017). The Hour of Code: Impact on Attitudes Towards and Self-Efficacy with Computer Science. Retrieved from https://code.org/files/HourOfCodeImpactStudy_Jan2017.pdf.

Banner photo by Visual Tag Mx from pexels.com

About the Author

Tisha Savannah is a graduate student in the MA Technical Communications program at Sam Houston State University. Tisha earned her BS in Professional Writing from the University of Houston-Downtown. Her interests include spending time with her family, reading, writing, and going to the movies.