

KNOWLEDGE QUEST

Journal of the American Association of School Librarians

AASL
American Association
of School Librarians
TRANSFORMING LEARNING

[KQ CONTENT](#)
[NEW!](#)
[BLOG TOPICS](#)
[COMMUNITY](#)
[About KQ](#)
[Contact KQ](#)
[Write for KQ](#)
[Blog for KQ](#)
[Advertise with KQ](#)
[KQ Archives](#)

[HOME](#) > [BLOG TOPICS](#) > COMPUTATIONAL THINKING IN YOUR SCHOOL LIBRARY

Computational Thinking in Your School Library

BY [DANIELLA SMITH](#) on [03/28/2019](#) · (0)



School and public libraries offer youth formal and informal learning environments, while providing access to a variety of resources. Librarians can introduce students to **computational thinking and coding concepts** in low-stakes contexts. This month's post features **Dr. Jennifer Moore** (associate professor in the School of Library and Information Studies at Texas Woman's University), Erin Bartley, Veronica Hernandez, and Amy Williams (three of her former students and current practicing librarians) discussing computational thinking (CT) in general and in the context of libraries.

Dr. Moore was selected to participate in Phase II of ALA's **Libraries Ready to Code** project. In this phase, ALA selected 6 faculty fellows to develop curricula to teach Master of Library & Information



Current Issue



Search

Science (MLIS) students how to implement coding programs in school and public libraries. After reading this post you can read more about the guest authors' experiences using these links:

- Erin Bartley's blog: <http://popplet.com/app/#/4291365>
- Veronica Hernandez's blog: <http://popplet.com/app/#/4295037>
- Amy Williams' blog: <http://popplet.com/app/#/4292335>
- Dr. Moore's article with other Phase II participants: <https://bit.ly/2Frsobe>

What is computational thinking (CT)?



Jennifer Moore (JM):

One cohesive definition of computational thinking does not exist. Multiple definitions exist, each with its own subtle nuances, and the intricacies of the definition depend upon the entity writing the definition. In my school library technology course, I share four

existing definitions: Jeannette Wing (2010), Google for Education (n.d.), International Society for Technology in Education/Computer Science Teachers Association (2011), and the ALA Office for Information Technology Policy (Braun & Visser, 2017), and then I created my own definition based on commonalities the definitions share: computational thinking involves an ordered (or systematic) problem-solving process that is transferable from computer science and K-12 education into other facets of our lives, including college, career, and everyday life issues.

Erin Bartley (EB): When I introduced this to my students, I described it as a process that we can use to approach problems that seem complex or "big." They related to that feeling of being overwhelmed when faced with a huge assignment, and we talked about how the steps of CT can make something complicated more



Most Popular

[Get Your Self\(ie\) Out There: My Year as an AASL Social Media Superstar](#)

[The 5 E's of Inquiry-Based Learning](#)

[Read Across America in the Middle School](#)

[Meet the 2020 AASL Candidates – Educators of School Librarians Section Chair-Elect](#)

[Tackling Senioritis through Engagement](#)

Recent Comments

[Sean Adcroft on Tackling Senioritis through Engagement](#)

[Robert Pattinson on 3 Reasons to Love Self-Checkout in an Elementary Library](#)

[Amy VanderMaas on Genrefication: Removing Barriers to Access](#)

manageable. I gave examples from my life to help them understand, like when I thought about reorganizing the library and coming up with a book fair schedule. When I called it a process for solving problems, many of the students made connections to the steps that they use to solve problems in math.

Veronica Hernandez (VH): Working with elementary students, I explain computational thinking as a system that we use to solve problems. Our campus focuses a lot on systems so that is a word that they are familiar with. Many times, a problem can seem so big that figuring out a way to break it into smaller pieces can be very helpful. Doing this can help the student see patterns that can also lead them to solutions that can be applied in different scenarios. I have used this with 4th grade. We talked about learning to look for the bones or skeleton of a problem. That can then be applied to math, science, or any other subject that they study. In other words, problem solving is not curriculum based but critical thinking.

Amy Williams (AW): When I explain this process to my students, I explain that it is a way of solving a problem by breaking it into smaller parts. Many times, a problem can be overwhelming and computational thinking is a logical way for them to break in to pieces so that it more manageable. When my students think of problem solving, they immediately think of a math problem. I explain to them then that computational thinking goes beyond math and can be used in all subject areas. I give examples of how I have used it in my life beyond school, such as when I am writing a knitting pattern, and how they might use it in their daily life. Once they see that this process is not limited to one subject, they are able to see many possibilities.

What is computational thinking not?

JM: One thing CT is not, however, is a discipline, nor is it specific to coding. Although CT, computer science, and coding are related, none of the terms are interchangeable. Computer science is a discipline. Coding is a skill set within computer science. Computational thinking is a problem-solving process and thus a foundational skill applicable to computer science and even coding specifically, but as previously mentioned, is applicable to other facets in life, too.

Why is computational thinking important?

Robert Pattinson on 2020 Vision – Goals for My School Library

Robert Pattinson on 12 More (mostly cheap) Teacher Tricks that Work in an Elementary Library

Sign Up For Email Alerts

Your email:

JM: Multiple problem-solving processes exist, so why CT? From an educational standpoint, educators should be designing lessons based on standards. The International Society for Technology in Education (ISTE) addresses computational thinking in both their Standards for Students and in their Standards for Educators (ISTE, 2019a; ISTE, 2019b). The American Association of School Librarians (2018) recently developed a document crosswalking the AASL *National School Library Standards* with the ISTE Standards for Students and Educators. In this crosswalk, AASL aligns computational thinking to multiple competencies for school librarians, the school library, and learners.

As previously mentioned, the development of CT skills transcends the academic setting. According to a report published by the American Library Association Office for Information Technology Policy (Braun & Visser, 2017), 58% of all new STEM jobs are in computing, 8% of STEM grads are in computer science, and 67% of computing jobs are outside the tech sector. According to Code.org, there are currently an estimated 557,903 computer jobs available nationwide, but only 42,291 computer science students who graduated in the workforce last year. Obviously, the demand for CS graduates far exceeds the supply.

Additionally, diversity in the computer science workforce is disappointing. In order to create better products that address the issues of more of the population, we need to diversify *who* becomes computer scientists. According to the National Center for Education Statistics (n.d.) and the National Science Foundation (2017), women of any ethnicity and people of color are largely underrepresented in the field. Google's multi-year research with Gallup Inc. revealed a range of complex barriers for groups underrepresented in CS (females, African-Americans, and Hispanics). These barriers start at a very early age and compound over time (Google-Gallup, 2015; Google-Gallup, 2016).

Of course, students learning CT is more than preparing a generation of people to be computer engineers and/or computer scientists. According to the World Economic Forum (2016), approximately 65% of primary school children are being prepared for jobs that currently do not exist. Because of access issues, many of our students are not developing the digital competencies and problem-solving skills preparing them for that future (Gallup, 2019).

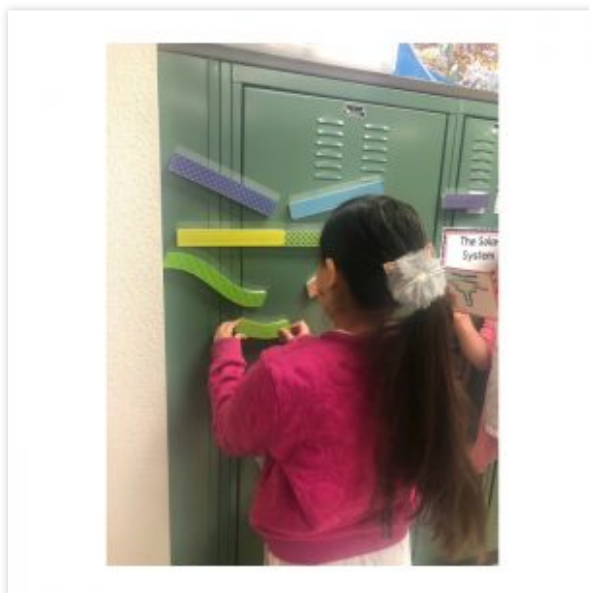
VH: Introducing students in early grades to computational thinking is very important because as stated above, they are being prepared for jobs that don't exist. Not knowing what these jobs will be, the best teachers can do is teach students how to identify problems and ways to then solve those problems with persistence. Those things are being left out in our curriculum. Many students are used to instant gratification that problem solving processes seem so overwhelming.

AW: When I was introducing Hour of Code to my students, we looked at the statistics of how many jobs we related to computer science and how fast that field is growing. It is so important to expose our students to problem-solving models that they will be able to use beyond high school. Our district has just opened a new STEAM center that not only is a campus for high school students to take classes, but also offers in-district field trips for every student in grades K-8 every year. Computational thinking is a perfect enhancement to what the students learn at the STEAM center.

EB: One of the things that you hear a lot in education is that critical thinking and problem solving is a challenge. Students might know a skill in isolation, but figuring out how to think through complex situations can get in their way. Providing early and consistent instruction in problem-solving processes, specifically CT, is a great way to give them a framework with which to think through some of those challenging, real-world problem scenarios.

What does computational thinking look like when applied to various problems, including coding?

JM: First, a bit of context: In 2017, I was involved with Phase 2 of the American Library Association (ALA) project, Libraries Ready to Code, funded by Google. Phase 2 involved six library and information science (LIS) faculty members, including me, from



across the nation collaborating to revise our graduate-level school library and youth services technology courses to integrate computational thinking (CT) and coding so that our students (future and current school and youth services librarians) will be better prepared to teach learners and faculty about both CT and coding. Students in my school library technology course created a flowchart, solving a self-selected problem by applying CT as a process using Google for Education's (2017) model (decomposition, pattern recognition, algorithm design, and abstraction) and then writing a rationale explaining their thought processes. The problem could be curricular/academic, personal, or fantastical. In a subsequent coding assignment, students had to discuss coding-specific problem solving using CT.

AW: At the time of this assignment, I was actually working on a project at home pertaining to knitting. I decided to use this as my computational thinking assignment. I had crocheted a round basket and a friend asked me to make her a square basket that would fit into some cubbies she had. I knew that I was not proficient enough in writing crochet patterns so I would need to write the pattern as a knitting pattern. There were many things I needed to consider such as needle size, yarn, gauge, and dimensions of the finished project. I also needed to create cables so that the overall effect would be the same as the crochet version. As I looked at the different components, I then needed to find patterns between the two. I then used an algorithm to create a knitting pattern for the new square basket making sure that my calculations were correct so the basket would fit into the cubbies. For the abstraction, I looked at ways to improve the basket such as making it sturdier by using 100% wool and felting the project, or changing the cable to make it more interesting. I also looked at doing market research to see if there would be a market to sell the baskets or pattern I created. Although this was not a school-related project, I have since found other ways to use computational thinking in my classroom and library.

EB: At the time that I worked on the assignment, I was a language arts teacher, and I was interested in applying CT to what my students were working on at the time: writing poetry. I imagined myself as a student struggling to write original lyrical poetry. I thought about how decomposition could help a student break down the writing task into manageable steps, such as selecting topics and prewriting, as well as how pattern recognition would help by

allowing students to notice how different types of poems tend to go. Algorithm design involved a step-by-step approach to writing lyrical poetry beginning with drafting and ending with planning where rhymes would occur in the lines of the poem. In abstraction, I brainstormed how this methodical approach would help with other writing and academic tasks.

VH: The assignment that I created focused on our elementary students bringing their own technology to school. At the time I was the technology teacher on a 1:1 campus where students were either assigned a district-owned tablet or had the choice of bringing their own. The problem then became accessibility. Students with district-owned technology had access to only district-approved apps while the students who brought their own technology had access to all the apps offered on their app store. When it came to periods of free use, students were upset because one had access to an app and the other didn't. In order to restrict access, the students who brought their own technology had to follow the district guidelines and use the district-approved apps when on school grounds. They still had access to their apps but were to be kept in a folder for use at home. With this restriction a solution that would make parents, teachers, and students happy was needed. We, a committee, then looked for alternative apps that were approved by the district and were similar to what was being downloaded by the students with a "free range app store." A list was created that showed the alternative apps like this: Instead of using *Piccollage* use *PicKids* or instead of using *Hopscotch* use *Scratch Jr*. This solution made everyone happy.

Why should librarians integrate computational thinking into their lessons and programming?

JM: Librarians are fortunate to teach in the largest inclusive classroom on campus and to interact with every student on campus, and library programs offer innovative and customized informal and formal learning spaces for all, thus improving access to skill development and resources for all learners. They teach technology skills and information literacy skills (among other myriad literacies), and through collaboration, they co-teach in other curricular areas. Because CT applies to not only coding but also to multiple curricular areas and real-life issues, librarians have the opportunity to integrate CT into multiple curricular areas, thus facilitating in the development of critical thinking skills and problem-solving skills.



Librarians have the opportunity to prepare their students for their educational careers, their professional careers, and their everyday life issues. It's a win-win situation.

VH: Librarians have a wonderful opportunity to teach CT everyday with the implementation of a makerspace. I have high tech and low tech in my makerspace, which provides students with opportunities to problem solve

and think critically while they “play.” Just today I sat and watched two students work on a circuit board and really problem solve the issues they were having. They were working collaboratively, thinking out loud, and problem solving. They even used the word debug! In my experience, not only are they able to learn new vocabulary but have an opportunity to put into practice how to problem solve with others. They are learning how to listen to each other, apply ideas, and debug!

AW: I think that libraries are a perfect place to incorporate computational thinking. Librarians are able to encompass all subject areas in their teaching and work with teachers to create lessons that include computational thinking. I am able to use it in all aspects of my teaching. I really enjoy collaborating with teachers to help them find ways to use it in their classrooms or how we can co-teach lessons using CT. The possibilities are endless using the process, and the true reward is seeing the growth that the students make when they are using this process.

EB: One of the wonderful things about teaching in the library is that more than any other place in the school, we are not perceived as being constrained by the boundaries of a particular subject area. On any given day, I might teach a reading skills lesson about nonfiction text features, and a lesson about research and database tools, and a lesson that involves coding and technology. We can apply the skills involved with computational thinking in all of these scenarios, and groups of students will get to see that whether they are working

in makerspaces, on a reading strategy, or on research, the same process can help them.

What does computational thinking look like for school librarians?



AW: I am very fortunate to be at a school where my library received a makerspace grant this year. The implementation of makerspaces has given me a wonderful opportunity to use computational thinking in my lessons. Each month I do a makerspace lesson with every class. The students are divided into groups and are given a problem that

they must work collaboratively to solve. I tie each problem with a book that is read before the lesson before introducing the problem they will be working on. In each scenario, the students work collaboratively using the CT model in order to solve the given problem. It has been amazing to see the growth that the students have made in less than a year. They are more confident in their work and eager to take on new challenges.

EB: CT can connect with so many different content areas, and as a result, it can be approached in lots of different ways. When students were first exploring the steps of computational thinking, we applied it to jigsaw puzzles that they were working on. At times, we specifically name the steps and talk about how abstraction or pattern recognition can help us in a certain situation. The steps are posted on a chart that students can refer to during their time in the library. At other times, as we are preparing to begin an assignment, I can ask them, “How can we use decomposition to help us out here?” If students are truly going to learn and apply CT, they need to be applying it regularly. It is almost like learning a new language—you have to hear it and use it to make it stick. One of my favorite things that came out of our introductory lessons this year was hearing students make connections about how CT can help them in

their lives. I asked them to add their ideas to a chart, and it was amazing to see their thinking. One group realized that they could use the steps to approach a big project that was coming up in math and science. Another connected it to playing video games, and another student shared that he could use CT to improve at his free throws in basketball. It is really a natural process, and once students learn about it and practice with it, the opportunities to apply are endless.

About the Guest Bloggers

Jennifer Moore is a former English language arts teacher and elementary school librarian and is currently an associate professor in the School of Library and Information Studies at Texas Woman's University. She teaches courses required for the School Librarian Certification, and her research interests include computational thinking in youth services and data literacy.

Erin Bartley is a TWU SLIS student who works as an elementary school librarian in Leander ISD.

Veronica Hernandez is a TWU SLIS graduate who currently works as an elementary school librarian in Lewisville ISD.

Amy Williams is a soon-to-be TWU SLIS graduate who currently works as an elementary school librarian in Allen ISD.

References

- American Association of School Librarians. 2018. *AASL National School Library Standards Crosswalk with ISTE Standards for Students and Educators*. Retrieved from <https://standards.aasl.org/wp-content/uploads/2018/08/180828-aasl-standards-crosswalk-iste.pdf>.
- Braun, L. & Visser, M. 2017. *Read to Code: Connecting Youth to CS Opportunity through Libraries*. Retrieved from <http://www.ala.org/advocacy/sites/ala.org.advocacy/files/content>
- Gallup. 2019. *Pioneering Results in the Blueprint of U.S. K-12 Computer Science Education*. Retrieved from <https://csedu.gallup.com/home.aspx>.
- Google. 2017. *Computational Thinking for Educators*. Retrieved from <https://computationalthinkingcourse.withgoogle.com/unit>.

Google for Education. n.d.. *CT Overview*. Retrieved from <https://edu.google.com/resources/programs/exploring-computational-thinking/#!ct-overview>.

Google-Gallup. 2016. *Diversity and Gaps in Computer Science: Exploring the Underrepresentation of Girls, Blacks, and Hispanics*. Retrieved from <http://services.google.com/fh/files/misc/diversity-gaps-in-computer-science-report.pdf>

Google-Gallup. 2015. *Searching for Computer Science: Access and Barriers in U.S. K-12 Education*. Retrieved from https://services.google.com/fh/files/misc/searching-for-computer-science_report.pdf.

International Society for Technology in Education. 2019a. *ISTE Standards for Students*. Retrieved from <https://www.iste.org/standards/for-students>.

International Society for Technology in Education. 2019b. *ISTE Standards for Educators*. Retrieved from <https://www.iste.org/standards/for-educators>.

International Society for Technology in Education & Computer Science Teachers Association. 2011. *Operational Definition of Computational Thinking for K-12 Education*. Retrieved from <http://www.iste.org/docs/ct-documents/computational-thinking-operational-definition-flyer.pdf?sfvrsn=2>,

National Center for Education Statistics. n.d. *Table 322.50. Bachelor's Degrees Conferred to Females by Postsecondary Institutions, by Race/Ethnicity and Field of Study: 2013-14 and 2014-15*. Retrieved from https://nces.ed.gov/programs/digest/d16/tables/dt16_322.50.asp?current=yes,

National Science Foundation. 2017. *Field of Degree: Women, Men, and Racial and Ethnic Groups*. Retrieved from <https://www.nsf.gov/statistics/2017/nsf17310/digest/fod-wmreg/whites-by-degree-share.cfm>,

Wing, J. 2010. *Computational Thinking: What and Why?* Retrieved from <https://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.p>

World Economic Forum. 2016. "Chapter 1: The Future of Jobs and Skills." *The Future of Jobs*. Retrieved from <http://reports.weforum.org/future-of-jobs-2016/chapter-1-the-future-of-jobs-and-skills/>.

April 2019 Professional Development

Organization	Date & Time	Professional Development Title
edWeb.net	April 9, 2019 – 4:00 pm – 5:00 pm EST	<u>How to Lead Conversations About Healthy, Responsible Media Use</u>
	April 10, 2019 – 4:00 pm – 5:00 pm EST	<u>How Early Literacy Impacts Reading to Learn: Research, Strategies and Digital Tools</u>
	April 11, 2019 – 3:00 pm – 4:00 pm EST	<u>Neuropsychological Evaluation and Treatment Issues in Children with Autism Spectrum Disorder (ASD)</u>
	April 15, 2019 – 3:00 pm – 4:00 pm EST	<u>Building Equity: Practices That Empower All Learners</u>
	April 17, 2019 – 3:00 pm – 4:00 pm EST	<u>Involve Parents for Greater English Learner Success</u>
	April 18, 2019 – 5:00 pm – 6:00 pm EST	<u>Authentic Assessment in the Digital Age</u>
	April 24, 2019 – 5:00	<u>Building a Schoolwide Culture of Reading</u>

	pm – 6:00 pm EST	
	April 25, 2019 – 3:00 pm – 4:00 pm EST	<u>Engaging Early Learners with Technology: Sharing Research and Best Practices</u>
	April 25, 2019 – 4:00 pm – 5:00 pm EST	<u>Educators and Social Media: Avoiding the Pitfalls</u>
	April 30, 2019 – 3:00 pm – 4:00 pm EST	<u>6 Steps to Create a Content- Rich K-12 Website</u>
Simple K12	April 6, 2019 – 10:00 am – 10:30 am EST	<u>Easily Create Websites, Movies, and More with Your Early Learners</u>
	April 6, 2019 – 11:00 am – 11:30 am EST	<u>STEAM Learning Through Play! Screen-free Coding & Robotics for Young Learners with KIBO</u>
	April 6, 2019 – 12:00 pm – 12:30 pm EST	<u>Formative Assessment Strategies and Tips for Early Childhood</u>
	April 6, 2019 – 2:00 pm – 2:30 pm EST	<u>Teaching the “T” in STEM: Technology for Young Learners (PreK-2)</u>
TeachersFirst	April 2, 2019 – 6:00 pm – 7:30 pm CST	<u>Game-Based Learning Basics</u>
	April 9, 2019	<u>Engage & Inspire:</u>

	– 6:00 pm – 7:30 pm CST	<u>BreakoutEDU Digital</u>
	April 30, 2019 – 6:00 pm – 7:30 pm CST	<u>Tech Integration Made Easy with Big Huge Labs</u>
InfoPeople.org	April 4, 2019 – 3:00 pm – 4:00 pm EST	<u>Celebrating Diversity & Multiculturalism through Crafts</u>
	April 9, 2019 – 3:00 pm – 4:00 pm EST	<u>Free Tools for Working with Graphics and the Web</u>
	April 10, 2019 – 1:00 pm – 2:00 pm EST	<u>Family Literacy Programming Goes to School: How our programs can help adult learners make a difference in the school lives of their children, from teacher conferences to homework help</u>
	April 17, 2019 – 3:00 pm – 4:00 pm EST	<u>School Readiness Webinar Series, Part 4: Overview of the School Readiness Toolkit</u>
VolunteerMatch.org	April 9, 2019 – 2:00 pm – 3:00 pm EST	<u>Engaging the Volunteer of the Future</u>
	April 16, 2019 – 2:00 pm – 3:00 pm EST	<u>Social Media and Volunteer Engagement</u>

	April 30, 2019 – 2:00 pm – 3:00 pm EST	<u>Successful Volunteer Interview Strategies</u>
School Library Journal	April 3, 2019 – 3:00 pm – 4:00 pm EST	<u>Game-Based Learning: Ideas for the Classroom & Library</u>
	April 24, 2019 – 3:00 pm – 4:00 pm EST	<u>Get Started With AR/VR/Mixed Reality</u>
	April 5, 2019 – 11:00 am – 12:00 pm EST	<u>Live Streaming Event with Margaret Peterson Haddix</u>
Texas State Library and Archives Commission	April 25, 2019 – 2:00 pm – 3:30 pm CST	<u>Get in the Game: Esports and Libraries</u>
American Association of School Librarians	April 3, 2019 – 6:00 pm CST	<u>The Power of Manga, Comics, & Graphic Novels through the Lens of the AASL Standards Frameworks for Learners</u>
	April 10, 2019 – 6:00 pm CST	<u>Addressing the Gatekeepers: How to Turn Comic and Graphic Novel Skeptics into Believers</u>
	April 17, 2019 – 6:00 pm CST	<u>Comics Librarianship: Essential Tools for the School Librarian</u>
Booklist	April 9, 2019 – 1:00 pm EST	<u>Middle-Grade Mania: Part One</u>

April 10, **Partner with School**
2019 – 2:00 **Educators Like a Boss**
pm EST

**Author: Daniella Smith**

Daniella Smith, PhD. is a former school and public librarian. She is currently an associate professor at the University of North Texas.

Share this:**Like this:**

Loading...

‹ **YoTeachApp – A New Backchannel**

Purposeful Instructional Technology and Formative Assessments in the Library ›

Categories: **Blog Topics** , **Collection Development** , **STEM/STEAM** , **Student Engagement/ Teaching Models** , **Technology** Tags: **coding concepts** , **computational thinking** , **google** , **Hour of Code** , **libraries ready to code** , **ready to code** , **rtc**

Leave a Reply

This site uses Akismet to reduce spam. **Learn how your comment data is processed.**

Order Back Issues
Subscribe

Contact KQ
Contact AASL

About AASL
Join AASL

META

Log in