

Computer Science Teacher Brief

The Computer Science Journey: Making a CS “Elective” Absolutely Essential

PROBLEM OF PRACTICE

In small and rural schools, computer science is often integrated into existing subjects (like Math, Science, etc.). The majority of the teachers who teach computer science also teach other content areas, with most teachers reporting that computer science was taught through the business, technology, or mathematics department in their schools. (Gal-Ezer & Stephenson, 2010).

Embedding computer science (CS) education into middle school curricula is essential for preparing students to innovate and solve problems in a technology-driven world that is increasingly shaped by computer technologies (Huang & Looi, 2021) but does not require educators to learn how to teach coding or navigate software-based curriculum and platforms.



Other schools incorporate computer science as an elective course... like in Pacheco Elementary School in Redding, CA. Brandi Holloway is teaching her third year of computer science (2025 - '26 school year) as an elective for 6th grade students. However, with a mix of such large ages and abilities, and a shortened window of time allotted each day (and by each trimester), Brandi found it challenging to provide equitable CS curricula for all her students.

This short brief introduces engaging ways that middle school educators can curate computer science electives, and offerings, to make CS work for their school, their students and themselves. Computer science is not an all or nothing curriculum... it is dynamic, fluid and can be customized to your individual needs – whether in populous urban areas or in more rural, smaller schools.

Pacheco Union School District – A Two School K-8 District

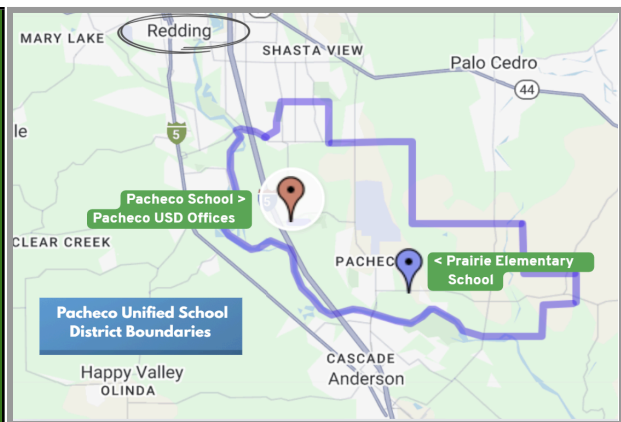
Pacheco Union School District is a public Tk-8 school district in Redding, CA (in the south side of the city), serving 720 students. The district consists of two schools: Prairie Elementary School,

servicing about 335 students in Grades Tk to 3; and Pacheco Elementary School is a 4th-8th grade campus with a total of 385 students.

Setting and Context

IN THE SPOTLIGHT: Brandi Holloway
6th grade teacher: Language Arts, Science
and Computer Science Elective
Pacheco Elementary School

SHASTA COUNTY.



Student demographics include: 69.1% Caucasian, 15.3% Hispanic, 4.2% American Indian or Alaskan Native, and then a combination of Black or African American, (1.3%) Asian (1%) and Filipino (0.3%). Pacheco Elementary is a Title 1 school with 3.3% of students identifying as homeless. There are 52.2% of students socioeconomically disadvantaged, with 2.2% identifying as English Learners.

School Motto: *Every Student Matters, Every Moment Counts*



What is Computer Science?

Computer science is a discipline, the study of computers, their applications, and their impact on society with a focus on understanding why they work and how to create those technologies (CCSIP, 2019). Computational thinking (CT) is a way to think about solving problems and draws on fundamental concepts of computer science (Wing, 2006) and is an essential 21st century skill and fundamental practice for young people to be successful in a digital world. For example, decomposition, pattern recognition, abstraction, algorithmic design, and evaluation are core CT competencies that are grounded in systemic approaches to problem analysis that cultivate student's problem-solving thinking skills and processes.

BRANDI'S CS JOURNEY

Brandi has participated in [Computer Science Discoveries curriculum](#) from [Code.org](#); in addition to the CS4NorCal/Seasons of CS Professional Learning Summer Institutes (2023, 2024 &

2025). Throughout the Pacheco School, Mrs. Holloway has taught 46-minute daily elective courses, including:

- > [Cyber.org](#) **Security Basics**: including **Cyberbullying, Personally Identifiable Information, and Strong Passwords**
- > **CS Discoveries**: focus on **Web Development, Interactive Animations, and Games**
- > **AI Machine Learning and Creating Apps with Devices** (Micro:bits)

*"I was finishing my induction program, and a flyer came across through an email about CS opportunities. I thought it would be something FUN to do over the summer. I ended up **loving it** and reached out to my principal before school started and let her know I wanted to do this as my elective for our students."*

– Brandi Holloway, Teacher Pacheco Elementary School



Many of our students have never used code before my class. I was taken aback by the students' enthusiasm for the course and eagerness to learn something new. Many of the students **expressed their enjoyment in the class** and wanted to learn more when they got to high school. Several students shared that the **problem-solving method** was helpful for them in other classes... especially math and science. They **accepted failure was part of the process** and worked to manage frustration when something did not work the first time.



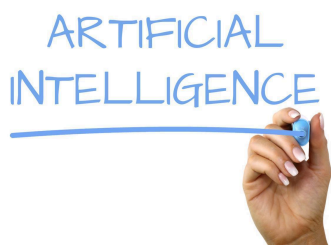
"I've taught CS for two full years as an elective for one period a day. During the first year, the students changed electives every trimester, and I struggled with picking the courses to work on during the trimester-long schedule."

This year will be very different for me with my elective students. Our school changed our schedule to reduce seven class periods to six; as a benefit, I now have 52 minutes with each class instead of 46 minutes. My elective students will change every trimester, and they will only get three days a week for their elective. The other two days are for WYN time. As my teaching CS time has significantly decreased, I am planning to start with security basics to promote safe online habits before starting CS Discoveries Interactive Animations and Games.

CHALLENGES

One of the most challenging aspects of teaching computer science was finding a way to meet the needs of the students in different grades in one class period. Having an elective of 6th, 7th, and 8th graders was challenging as they have different background knowledge, focus issues, maturity, and reading skills.

Brandi shared how she “also found it challenging to find the right resources and tools to help understand how Ai is affecting our future. Our students are aware of Ai, but not how it explicitly works and how they should be using it. This also applies to technology in general. Students are aware of it, but not necessarily how it works in school and in their personal lives.”



I am also using the *Everyday AI* curriculum in my language arts classes to help students understand how AI works and help them learn that these are resources available to help them build skills. I'm working to debunk the idea that ChatGPT is just for copying and pasting information; when used effectively, ChatGPT can be a great educational resource.

PROMISING APPROACHES

From Trimester Sessions to Year-Long Program

A significant change in our computer science program was transitioning the elective from a trimester to a year-long course. This switch allowed for deeper learning and the development of a more cohesive classroom community. The impact was immediate and profound; after the elective duration change, Brandi had over 90 kids request her class – triple the amount she could accommodate (25).

Actionable Advice for Educators

I recommend having a CS elective towards the end of the day, depending on the student's grade level. For example, sixth graders are typically 11-years-old and are already struggling with changing classes every period and being in junior high. Their stamina is being tested and having an engaging class towards the end of the day typically shows better results. As you get to know your students, you can customize the CS content to their real lives and interests (such as: social media algorithms, mobile apps, and video games). There are life skills in every CS

course that include pair programming and group projects, which build communication and teamwork skills.

- **Unit 1: Cybersecurity and Online Safety**, 2 weeks
(safe passwords, cyberbullying, personal info.)
- **Unit 2: What is a computer?**, 2 weeks
(parts of a computer, how they work)
- **Unit 3: Game Lab**, 2 weeks
(creating shapes, changing colors, random numbers, sprites)
- **Unit 4: Web Design**, 3 - 4 weeks
(creating small & large projects, adding pictures, colors, fonts, etc.)

RECOMMENDATIONS

For other teachers looking to build or enhance a middle school CS program, Brandi offers some practical advice for the classroom, based on her teaching experiences.

Make it Relevant

"As you get to know your students, you can customize the CS content to their real lives and interests, such as social media algorithms, mobile apps, and video games."

- Maximizing CS content (e.g., identifying which CSD units to use, using CSI lessons)
- Find what best works for you and your students. As the teacher, we know our students best.
- You don't have to teach every lesson in the CS curriculum.
- Use rubrics for projects (clarity, correctness, creativity, documentation). Scaffold lessons for beginners, intermediate, and advanced coders.
- I instruct each student to do 2 practices and 1 challenge to show completeness of the lesson; however, those who finish early are asked to do more challenges.
- Don't be afraid to put the answer key on the board for students to look at after they have tried and failed several times. That's when the aha moments occur because they see the simple error they made and can move on with confidence.

Anyone can teach CS!

Embracing "Unplugged" Learning

Brandi is a firm believer in "unplugged" lessons – an approach where students engage with computational challenges away from a device to understand key principles. (Bell, 2021).



“One of my favorite parts of CS Discoveries is the unplugged lessons. It reinforces technology without being on a Chromebook. It is an integral part of the learning process for young students.”

Brandi also values the journaling component of the Code.org curriculum... which encourages students to disconnect, write, and reflect on their learning.

THINGS TO CONSIDER

Key Learnings and Reflections

Every class typically has 1-2 students who have never coded before and pick up the concepts so easily. They become the class resource to help others. Often, the *Girls Who Code* are more confident and work better together to problem solve. I absolutely love seeing a student put their hands up in triumph because they worked the problem-solving method to figure out and fix their errors.

I have been working with our local high school to try to coordinate a field trip to see other CS courses. During our next district meeting, I would like to discuss bringing more computer science options to our lower grades. There is a drone competition coming up that I am going to attend this year with students.

Vertical articulation is the process of aligning curriculum and instruction across different grade levels to create a logical and cohesive learning progression for students. It involves teachers collaborating to ensure that each year's content builds on the previous one, preventing learning gaps and promoting deeper understanding as students advance through their education. For example, a vertically articulated math curriculum would ensure a solid foundation in multiplication is in place before moving on to more complex concepts like fractions or algebra in later grades.

RESOURCES

[CS4NorCal](#) – A grant-funded project that provided CS training and resources for teachers. All resources are not too low-cost and offer year-round workshops and support.

[CSTA California Far North Chapter](#) – Established as a CS Community of Practice to provide ongoing CS training, resources, and tools and to help support and connect all teachers interested in and teaching computer science.

[Code.org](#)'s CS Discoveries – An introductory computer science course that empowers 6th through 10th grade students to create authentic artifacts and engage with computer science as a medium for creativity, communication, problem solving, and fun. CS Discoveries can be taught as a 1 or 2 semester course (or 50+ course hours).

Topics included in the curriculum:

- Artificial Intelligence
- App Design
- Art and Design
- Data
- Games and Animations
- Physical Computing
- Programming
- Web Desi

REFERENCES

California Computer Science Strategic Implementation Plan (CSSIP), 2019. Retrieved: <https://www.cde.ca.gov/pd/ca/cs/documents/cssiplegreport2019.docx>

Code.org, CSTA, ECEP Alliance (2023). 2023 State of Computer Science Education: Retrieved from <https://advocacy.code.org/stateofcs>

K–12 Computer Science Framework. (2016). Retrieved from <http://www.k12cs.org>.

Mills, K., Coenraad, M., Ruiz, P., Burke, Q., & Weisgrau J. (2021, December). *Computational thinking for an inclusive world: A resource for educators to learn and lead*. Digital Promise. <https://doi.org/10.51388/20.500.12265/138>

Saw, G. K., & Agger, C. A. (2021). STEM Pathways of Rural and Small-Town Students: Opportunities to Learn, Aspirations, Preparation, and College Enrollment. *Educational Researcher*, 50(9), 595-606. <https://doi.org/10.3102/0013189X211027528>

Showalter, D., Hartman, S.L., Eppley, K., Johnson, J., & Klein, R. (2023). Why rural matters 2023: Centering equity and opportunity. National Rural Education Association.

AUTHORED BY:

Brandi Holloway, 6th grade teacher (English, Language Arts, CS electives) Pacheco School
Jonathan Rhodea, Computer Science Curriculum Specialist, Sacramento COE

Computand sponsored by the Small School Districts' Association. er Science in Rural CA (CS4NorCal) is a project funded by a U.S. Department of Education grant