# The Advocate

Volume 27
Number 1 *Fall-Winter*

Article 4

November 2021


# Teaching Computer Science: An Exploration of Habits of Mind


Lauren Hays
*University of Central Missouri*, lauhays@ucmo.edu

Jenna Kammer
jkammer@ucmo.edu





## Recommended Citation

Hays, Lauren and Kammer, Jenna () "Teaching Computer Science: An Exploration of Habits of Mind," *The Advocate*: Vol. 27: No. 1. https://doi.org/10.4148/2637-4552.1164

# Teaching Computer Science: An Exploration of Habits of Mind


## Abstract

Previous research has indicated that many K-12 computer science teachers do not have a background in computer science education. This means, computer science teachers are not necessarily using established habits of mind they learned from their own studies. The purpose of this study is to understand more about the habits of mind needed by computer science educators, and students, who often have varying levels of ability and knowledge of computer science.

We answer the following research questions:

1. How do teachers of computer science use teaching strategies to develop the habits of mind needed for computing?
2. What habits of mind do teachers of computer science name as important for teaching?
3. What habits of mind do teachers of computer science name as important for learning?

To answer the research questions, we used narrative analysis. Findings showed that the backgrounds of computer science teachers greatly impacted their pedagogy and the habits of mind used.



## Keywords

habits of mind, computer science education, teaching strategies, pre-service teacher programs

# Teaching Computer Science: An Exploration of Habits of Mind


## Lauren Hays, University of Central Missouri
*Lauren Hays is an assistant professor of instructional technology at the University of Central Missouri, Warrensburg, Missouri.*

## Jenna Kammer, University of Central Missouri
*Jenna Kammer is an assistant professor of library science at the University of Central Missouri, Warrensburg, Missouri*


### Introduction

Computer science education in K-12 schools has expanded swiftly in the last several years as states adopt new computer science standards. Most states in the United States now have computer science education standards in place with more currently in progress (Code.org, CSTA, & ECEP Alliance, 2020). Kansas adopted their most recent computer science standards in 2019 (KSDE, 2019). While some states have also allocated funding to train teachers to become certified in computer science, many states are still developing policies that will allocate funding to training and certifying computer science teachers.

Our interest in this topic arose as educators of school librarians and technology coaches. We noticed that our graduate students were increasingly asked to teach computer science in schools, and were chosen for their ability to learn and use technology (a habit of mind) rather than because of a formal computer science background. Previous studies indicate that many computer science educators come from other content areas with little experience in computer science, or they have a background in industry with little experience in teaching (Rich et al., 2019; Shulte et al., 2012; Yadav, Gretter, Hambrusch, & Sands, 2016). Computer science has also not been traditionally taught as a content area in many teacher education programs (Shulte et al., 2012). Because of this, computer science educators will often be self-taught, seek professional development within their communities, and change teaching practices when they start teaching computer science (Sentance & Csizmadia, 2016). These challenges can be difficult for both teachers and students of computer science.

We, the researchers, identified one solution to the challenges faced by computer science educators, as  developing habits of mind that acknowledge computer science learning as a continual and lifelong learning process. The rationale for this solution was derived from our own experience in education. We knew that certain mental habits are needed to engage in learning and address unexpected and unfamiliar educational settings. Costa (n.d.) wrote, “when humans experience dichotomies, are confused by dilemmas, or come face to face with uncertainties—our most effective actions require drawing forth certain patterns of intellectual behavior” (para 2). For many computer science teachers, uncertainty defines their teaching experiences (Ni et al., 2021). Additionally, since we believed habits of mind were a solution to address difficulties faced by computer science teachers, then, we also felt it was important to understand what habits of mind teachers sought to cultivate in their students.

The purpose of this study is to understand more about the habits of mind needed by computer science educators, and students, who often have varying levels of ability and knowledge of computer science. Specifically, we sought to understand what habits of mind computer science teachers believe are needed to teach computer science, and what habits of mind computer science teachers believe are necessary for their students to learn computer science. As teacher preparation programs add computer science as a subject area, knowing what habits of mind are needed by teachers and their students will be beneficial.

Through extensive interviewing with practicing computer science educators, this pilot study involved ten computer science educators with varying backgrounds and professional titles to learn more about the habits of mind they use and encourage. The findings answered the following research questions:

1. How do teachers of computer science use teaching strategies to develop the habits of mind needed for computing?
2. What habits of mind do teachers of computer science name as important for teaching?
3. What habits of mind do teachers of computer science name as important for learning?

In this paper, we specifically use the phrase "teachers of computer science" to encompass all participants. When recruiting participants, we found that some teachers who teach computer science courses or programs may have another primary role in other parts of the school. For example, participant A was a school librarian who taught computer science concepts to students during specific units of the curriculum. Participant B is the business teacher who also teaches computer science in the high school.

## Literature Review

### Computer Science Education in K-12

#### *Expanding curriculum*

Computer science education in K-12 schools in the United States has increased as new states adopt computer science standards. Code.org (2019) (an advocacy group for computer science education in the United States) reports that standards are now adopted in the majority of states. These standards, often lobbied for by private organizations (like Google) and nonprofits (like Code.org), was the initial move to embed computer science into the K-12 curriculum as a content area in the core sciences, rather than a fringe or elective subject (Nager & Atkinson, 2016). Webb et al. (2017) explained that discussions about the expanding computer science curriculum have been happening globally since 2012, with general consensus that is needed in the K-12 curriculum, though with varying degrees of interpretation.

*Teacher gap*.

Finding teachers for computer science has proven challenging for many schools, often requiring schools to move teachers into computer science roles. Yadav et al. (2016) found that teachers


DOI: 10.4148/2637-4552.1164

who were asked to teach computer science felt that they did not receive adequate professional development to be able to teach the content, often requiring teachers to learn the content on their own. Therefore, many schools have invested in providing professional development to existing teachers to prepare them for the role of computer science teacher (Qian, Hambrusch, Yadav, & Gretter, 2018). Qian et al. recommended that professional development for computer science educators should include matching professional development with the teacher's background, aligning professional development with the curriculum, and using effective motivational design. Important to note is that professional development varies by country and national investment in computer science education. For example, Rich et al. (2019) found that teachers in the U.K. were significantly more experienced in formal computer science education than K-8 educators in other countries. Rich et al.'s research also found that computer science teachers felt like there were many challenges to teaching this subject, including not having extensive background knowledge in computer science, keeping up with student knowledge and solving problems that occur in student projects.

### *Teaching strategies*

Teaching computer science is a constructivist approach that involves active learning strategies, including problem solving, critical thinking and authentic learning experiences (Sentance & Csizmadia, 2016). Specifically, Sentance and Csizmadia (2016) found that teaching computer science required educators to change teaching strategies, develop computational thinking and resilience. Gal-Exer and Stephenson (2014) also found that there are cross-cultural challenges with teaching computer science such as finding an engaging curriculum for all students, and ensuring that the teachers have the technical knowledge to teach computer science. Teachers must also adapt teaching strategies to the hardware, software and resources that are available within their school.

Teachers may also have specific strategies that work well for teaching computer science. Carbone and Kaasbøll (1998) found that the most common methods for evaluating the success of teaching innovations in computer science were through strategic problem solving, iteration (a process of continual repetition and revision), interfacing (testing projects with others). Ismail, Ngah, and Umar (2010) also found that effective teaching strategies for teaching computer science should include beginning with structured or procedural programming languages, and ensuring that students are actively involved during programming practice. They conclude that active learning can help students to learn at a higher level.

However, many computer science teachers do not have experience taking formal computer science courses themselves. Due to this, they cannot rely on using strategies they saw modeled in their own education. Instead, they must rely on industry experience, professional development, and professional learning networks for pedagogical ideas.

### *Habits of Mind*

According to LeBlanc (n.d.) "A habit of mind is a usual way of thinking, a way of engaging with the everyday world" (Habits of Mind, para. 1). Habits of mind impact how individuals perceive their circumstances and how they react to their environment; they impact how students approach learning and how teachers approach teaching.

Gordon (2011) wrote about the need to develop mathematical habits of mind in students and pre-service mathematics teachers. Specifically, he indicated a need to encourage students to reflect on what they have done and on what they have not. The importance of mathematical habits of mind is echoed by Matsuura, Sword, Piecham, Stevens, and Cuoco (2013) and Umar (2017). Similarly, the Framework for Success in Postsecondary Writing identified eight habits of mind that are necessary for success in college writing (Council of Writing Program Administrators, National Council of Teachers of English, & National Writing Project, 2011). These habits of mind are:

- Curiosity – the desire to know more about the world.
- Openness – the willingness to consider new ways of being and thinking in the world.
- Engagement – a sense of investment and involvement in learning.
- Creativity – the ability to use novel approaches for generating, investigating, and representing ideas.
- Persistence – the ability to sustain interest in and attention to short- and long-term projects.
- Responsibility – the ability to take ownership of one's actions and understand the consequences of those actions for oneself and others.
- Flexibility – the ability to adapt to situations, expectations, or demands.
- Metacognition – the ability to reflect on one's own thinking as well as on the individual and cultural processes used to structure knowledge. (Council of Writing Program Administrators, National Council of Teachers of English, & National Writing Project, 2011, p. 1)

Computational thinking is widely discussed in computer science education literature as a skill needed for teaching and learning computer science (Guenaga, Eguíluz, Garaizar, & Gibaja, 2021; Yadav, Krist, Good, & Caeli, 2018). However, the researchers were unable to locate any literature describing other habits of mind needed within computer science. Because computer science has not identified a definitive list of habits of mind for the field, the researchers chose to use a list of 16 habits of mind developed by Costa and Kallick (2008) because they have been applied to other content areas in K-12 education, and are often used across the full curriculum and are transferable between schools (Costa and Kallick, 2009) . Costa and Kallick's (2008) list included habits of mind identified in other K-12 content areas (i.e. Council of Writing Program Administrators, National Council of Teachers of English, & National Writing Project, 2011) and included additional habits the researchers believed may be perceived as important by computer science teachers. Kallick and Costa (2021) defined habits of mind as "dispositions that empower creative and critical thinking" (para. 3). In their book, *Learning and Leading with Habits of Mind: 16 Essential Characteristics for Success*, Costa and Kallick (2008) listed 16 mental habits. These habits of mind are:

- Persisting
- Managing impulsivity
- Listening with understanding and empathy
- Thinking flexibly
- Thinking about your thinking (metacognition)


DOI: 10.4148/2637-4552.1164

- Striving for accuracy
- Questioning and problem posing
- Applying past knowledge to new situations
- Thinking and communicating with clarity and precision
- Gather data through all senses
- Creating, imagining, innovating
- Responding with wonderment and awe
- Taking responsible risks
- Finding humor
- Thinking interdependently
- Remaining open to continuous learning

While not exhaustive, these 16 habits of mind serve as a framework for understanding how individuals process information, work through new knowledge, and approach unfamiliar problems.

## Methods

The researchers chose an exploratory approach to understand more about the habits of mind needed to teach computer science in K-12 schools because there is little research published in this area. Specifically, the researchers used narrative analysis to understand the stories of computer science teachers and how those stories impacted their views on the habits of mind necessary for students learning coding and instructors teaching coding. Narrative research provides a method to share the stories of the participants, from their perspective and in the way that they see their job. Therefore, by using narrative analysis, the researchers gave voice to the experiences of K-12 educators (see Table 1 for participant demographics) who are paving the way in computer science and often have to learn on the job through trial and error (Clandinin & Connelly, 2000).

Table 1. Demographics

| Participants | Job Title | School Level | Previous Experience |
|---|---|---|---|
| A | School Librarian | Elementary | Social Studies Teacher |
| B | Business Teacher | High | Programmer |
| C | Gifted Education Teacher | Middle | Special Education Teacher |
| D | Computer Science Teacher | High | Business and Computer Science Teacher |
| E | Computer Science Teacher | Middle-High (6-12) | Robotics Instructor |
| F | Math Teacher | High | Math Teacher |

| | | | |
|---|---|---|---|
| G | Computer Science Teacher | High | Math Teacher |
| H | Computer Science Teacher | Middle- High (7-12) | Science Teacher |
| I | Business Teacher | High (10-12) | Professional Marketer |
| J | Instructor of Computer Science and Digital Photography | High | Art teacher |

Interviews were conducted and recorded on Zoom with each participant, totalling over 448 minutes of interviews. During the interview, participants were asked to share the story of how they became computer science teachers. In the second half of the interview, participants reviewed Costa and Kallick's (2008) habits of mind, and shared stories as to how the habits of mind were relevant to their teaching and learning practices. Interviews were transcribed and coded for the habits of mind referenced in the interviews, as well as details related to the beginning, middle and end of their stories as computer science educators. Each habit of mind serves as a start code. To ensure the stories were accurately represented, the researchers engaged in member checking by having the participants review the transcripts. The participants all confirmed that their voices and stories were accurate and present in the transcripts.

Once member checking was complete, the researchers wrote summaries of the participants' stories clearly identifying a beginning, middle, and an ending for the participants' current journeys as computer science educators (Clandinin & Connelly, 2000). After the stories were written and a beginning, middle, and end, for each story was established, the researchers coded the interviews, using the habits of mind identified by Costa and Kallick (2008) as the framework. Costa and Kallick's (2008) list of 16 habits of mind was selected because they are commonly agreed upon characteristics of effective behavior for solving problems. Further, faculty in teacher education programs will recognize many of the mental habits as ones they already seek to cultivate in their students.

## Findings

### Stories of participants

#### *Participant A*

Before beginning a career in education, participant A was an office manager for a modem distributor computer modem distributor. Once she began in education, she started as a social studies teacher before moving into her current position as a school librarian which she has been in for six years. These varied experiences helped her grow and become more comfortable with technology. Also, she has many family members who work in IT and their experiences impact the technology she is able to experience. For her, technology connects people even though many people see it as a disconnector. Five years ago, she attended professional training on Code.org and immediately implemented it by having students participate in Hour of Code. Each year, she continues to work with Hour of Code and has now blocked off the months of November and December to be designated coding months.

### *Participant B*

Participant B began her career in industry and worked as a programmer installing systems. After that, she got into IT audit. After 15 years in industry, she moved to the university level to teach computer science which she did for 10 years. Currently, she works as a business teacher teaching computer science to grades 9-12 at a suburban high school. Her background in computer science led her to rework her courses from a year-long format to individual semesters in order to give more students the opportunity to take a computer science course. Additionally, her connections to industry and higher education led her to drop a networking course and add a computer architecture course. She has spent a lot of time working to ensure her students obtain the skills they need to move forward in the field after graduation.

### *Participant C*

Participant C worked in sales before beginning a career in education. At age 30, she began her teaching career in special education and then began teaching gifted education. When sixth grade students in her district needed an extra class, she offered to teach a computer science course. Her computer science course is now offered to all sixth grade students with the exception of those in band. She believes her background in sales has impacted her teaching. She approaches her instruction with a focus on communication and collaboration.

### *Participant D*

Participant D began his career as a business teacher before becoming a computer science teacher. He has been a teacher for 23 years and for the past 15 years he has taught computer science. He spent a few years as a district technology coordinator before moving back to teaching full-time. He learned computer science on his own and through a minor he obtained in college in computer information systems. Currently, he teaches 10th, 11th, and 12th grade computer science at a suburban high school. Specifically, he teaches computer science principles, computer science A, computer science essentials and cyber security, which are all Project Lead The Way courses (Project Lead The Way is a non-profit organization that develops STEM curricula for K-12 education). To stay up-to-date on computer science he reads forums and solves computer science problems such as the Advent of Code.

### *Participant E*

Participant E began his career as a math teacher. He taught math for 10 years before starting to teach technology and robotics courses. Seven years ago, he started teaching Scratch. For the past two years, he has been a computer science teacher for grades 6 through 12, and continues to be the robotics coach. He has taught Python, C++, Java. HTML, and CSS. He learned computer science through college electives and through participating in Global Hack. Despite having a master's degree in computer technology, he is only temporarily certified to teach computer science.

### *Participant F*

Participant F began his career as a mathematics teacher where he taught Geometry, pre-calculus, trigonometry, and algebra 1. During his second year, he introduced coding as an elective and the program took off. In the following year, he had two sections of the elective coding course and by his fourth year teaching he was teaching two sections of elective coding plus AP Computer Science principles. Now, in his fifth year, his full teaching load is computer science courses with the exception of a geospatial information systems course. While he no longer has any math courses in his teaching load, he remains in the mathematics department. During college, he took one computer science course that was part of his math degree, but beyond that has not had any formal training. He has learned computer science by teaching the courses, workshops, and taking a class on Udemy. He describes himself as primarily self-taught. Despite this lack of formal background in computer science, he has been extremely successful in promoting computer science at his school and students in computer science internships and pursuing degrees in computer science.

### *Participant G*

A former math teacher, Participant G began teaching AP Computer Science eight years ago. During college, he obtained a math degree with a minor in computer science so he had some familiarity with computer science prior to teaching the AP course; however, he said it had been six years since he had done any programming when he agreed to teach the course. To refresh his skills and stay up-to-date, he teaches himself through tutorials and online resources. Today, he teaches computer science Project Lead the Way courses, is a master teacher with Project Lead the Way, and is on the Project Lead the Way National Advisory Board. He describes himself as very passionate about computer science education and wants to see it continue to improve.

### *Participant H*

Participant H began his career in the military, then studied plant biology during his undergraduate degree. He worked various jobs, received a graduate degree in biology, and then studied for a master's degree in teaching. Previously, he taught high school science and ran a coding club. From the work with the coding club, he moved into teaching computer science, and currently he teaches 7th- 12th grade computer science. In his current position, he teaches computer science essentials, cybersecurity, and app creators which are all Project Lead the Way courses. He teaches classes that are not part of the Project Lead the Way curriculum as well.

### *Participant I*

Participant I received a bachelor of science in marketing and business administration and then worked for many years in the business world as a stock broker and then as a food broker. He spent many years traveling the world, and selling food. A downturn in the industry led him to seek out other career opportunities. He had always been interested in teaching, so he applied for a business teaching position at a career center in a rural town. He convinced the individuals interviewing him that his experience in the business world would be invaluable. During the interview he was asked if he could teach computer programming, and he said he would learn. The summer before he began teaching, he spent a great deal of time talking to software developers and learning what he needed to know. Currently, he teaches sophomores, juniors, and seniors computer science. He is always learning something new in order to help his students grow in their knowledge because his students are very important to him.

### *Participant J*

Participant J has a master's degree in curriculum and instruction. Her teaching background is art. She became interested in coding when her children were interested in technology and robotics, but the school did not have a robotics team. She did not know how to do robotics, but stepped up and became the robotics coach. Then, when her children were in high school there the school did not have any technology courses available so she got a Project Lead the Way engineering program going. In her art classes, she teaches Adobe Photoshop and InDesign, so is comfortable on the computer; however, computer science is not her background. After the success of the Project Lead the Way engineering program, it was determined that the school needed to start a Project Lead the Way computer science program. She decided to take on the challenge and received the training she needed. Her current job title is instructor of computer science, digital photography, and the yearbook. In this role, she is always learning and has worked with a computer science professor at a local university to obtain the knowledge she needs.

## Participant Backgrounds

In order to make sense of each participants' story and see common themes, the researchers grouped participants in categories based on their backgrounds (see Table 2). A background in math and science was the most common area from which computer science teachers came; however, 50% of the study population came from a background other than math or science education. This finding was in line with results from other studies (Rich et al., 2019; Yadav et al., 2016).

Table 2. Participant Categories

| **Category** | **Background** | **Participants** |
|---|---|---|
| Industry | Computer science industry | Participant B |
| | Business industry | Participant I |
| Education | Education--math or science | Participant D<br>Participant E<br>Participant F |

| | | |
|---|---|---|
| | | Participant G<br>Participant H |
| | Education-Other | Participant A<br>Participant C<br>Participant J |

**Research Question 1**

*How do teachers of computer science use teaching strategies to develop the habits of mind needed for computing?*

The study participants were asked to describe how they teach computer science to their students to develop the habits of mind needed to be successful at computer science. Twenty-two codes were identified and categorized into 6 themes. The study participants indicated that they provided opportunities to work together (including pair programming or creating teams), assign iterative projects (including project based learning that connects with real world problems or experiences), integrate digital literacy (including teaching about academic integrity, copyright, and searching for needed information), build excitement for content (including allowing for working in stations, provided choices of projects, bringing in experts to talk to the class, or assigning game based learning), empower students at various levels (including addressing the ability gap through instruction and ensuring equal opportunities for various genders, race and socioeconomic backgrounds), and integrating active learning (including limiting instructor-led content in favor of self-discovering and problem-solving) (see Table 3).

Table 3. Teaching strategies paired with habits of mind assigned by researchers

| Theme | Codes | Evidence | Related Habit of Mind |
|---|---|---|---|
| **Opportunities to work together** | Collaboration<br>Teamwork<br>Pair Programming<br>Discussion | "I start off with pair programming. They have a navigator and driver. The driver is the one actually doing the programming and the navigator is the one looking for mistakes, looking things up and communicating with me if there are problems." (Participant H) | Thinking interdependently |
| **Assign iterative projects** | Agile Programming<br>Planning Tools<br>Scaffolding<br>Sequence, decision, repetition | "The agile programming method breaks it down into smaller parts. Then they get it working, move on to the plan and make sure to add the next part to then get that working and add the next part. It's kind of an iterative process, which is also what | Applying past knowledge to new situations |

| | | | |
|---|---|---|---|
| | Design Thinking | the majority industry uses.” (Participant D) | |
| **Integrate digital literacy** | Information ethics<br>Academic integrity<br>Vocabulary | “The other thing that I do in all my classes is vocabulary. They all have weekly vocabulary quizzes over terms. I really stress the language so that when we're using it in class, they understand what we're referring to.” (Participant B) | Thinking flexibly |
| **Build excitement for content** | Stations<br>Connect to Real World<br>Engaging Experts<br>Games<br>Choice<br>Self-paced | “I will set up all the different stations and then they will go to those stations and explore. It's a lot of self-guided, a lot of growth mindset. So they fail. They get frustrated. They raise their hand. I come over. I give encouraging words and say, ‘Hey, why don't you try just doing this’ and then they fix it themselves, and they feel that success.” (Participant A) | Responding with wonderment and awe |
| **Empower students at various levels** | Choice<br>Self-paced<br>Recognize Ability Gap<br>Diversity, Equity and Inclusion efforts | “It's just about finding the right level of challenge for every kid. I have kids walking in and their parents are developers and they have been programming since they were five, and then I also have kids and walk in this is the first day they've ever seen any code whatsoever. What's cool about the open-ended stuff and those kind of projects is that when you say ‘hey build something that's relevant to you’ or ‘build something that you're interested in’ what success looks like for each of those kids is very different. But it's all allowable in the same classroom.” (Participant G) | Taking responsible risks |
| **Integrate active learning** | Learning from Mistakes<br>Microlectures<br>Collaboration | “At the beginning of the three-hour block we would spend somewhere between probably 15 minutes to 20 minutes covering a new topic. I try never ever to lecture more than 20 minutes of our three-hour block.” | Thinking and communicating with clarity and precision |

(Participant I)

**Research Question 2**

*What habits of mind do teachers of computer science name as important for teaching?*

Study participants identified 14 of 16 habits of mind as being important for teaching computer science. The most frequently mentioned habits of mind were taking responsible risks (n=5), finding humor (n=4), and remain open to continuous learning (n-4). This range of opinions about the mental habits needed for teaching computer science underscores the various backgrounds of the educators themselves. Many study participants did not have a background in computer science, but were introduced to computer science through professional development. Table 4 showcases the habits of mind identified by the study participants, the number of participants that named the habit of mind as important, the specific participants that named the habit of mind, and an example of one participant's discussion of the habit of mind.

Table 4. Habit of mind coding teachers need as described by the participants

| **Habits of Mind** | **Number of Participants** | **Participants** | **Example Evidence** |
|---|---|---|---|
| Taking responsible risks | 5 | Participant B<br>Participant C<br>Participant D<br>Participant H<br>Participant J | "Taking responsible risk. They are all good, but sometimes those responsible risks, because sometimes you will have someone really interested in something or what not and you may not be entirely familiar with it but sometimes you just have to go with it...and not be afraid to make mistakes." (Participant H) |
| Finding humor | 4 | Participant B<br>Participant E<br>Participant F<br>Participant J | "You have to have that humor because learning is uncomfortable and scary and I am uncomfortable learning computer science." (Participant J) |
| Remain open to continuous learning | 4 | Participant A<br>Participant D<br>Participant G<br>Participant H | "...constantly reading forums, trying to solve problems as they come out. And do stuff like there's some websites that like tracking code, some of those problems. One of the biggest ones that I like is the |

| | | | |
|---|---|---|---|
| | | | Advent of Code." (Participant D) |
| Creating, imagining, innovating | 3 | Participant A<br>Participant F<br>Participant I | "I need to always be saying, Okay, this is what we do. What's the better way to do it, what's a faster way to do it. What will make them excited to do it?" (Participant I) |
| Persisting | 2 | Participant A<br>Participant D | "I mean, we had a bunch of librarians and other technology people in the room, and we were all just like, this is really hard. How are we ever getting our kids to do this, but just persistence and keep going through and we did collaborating."(Participant A) |
| Thinking flexibly | 2 | Participant F<br>Participant H | "You have to change perspectives on what you're used to. And in a lot of times for a lot of teachers, teaching coding means to pick up a new practice." (Participant F) |
| Striving for accuracy | 2 | Participant D<br>Participant E | "I think is has to be exact...you need to be able to show that you're gonna make mistakes too. You don't have all the answers." (Participant D) |
| Thinking and communicating with clarity and precision | 2 | Participant B<br>Participant E | "People hear what they want to hear, so you have to say it in such a way that everybody hears the same thing. I'm striving for accuracy, and questioning and posing problems on getting the kids engaged. They don't want to deconstruct things that are already there." (Participant E) |
| Questioning and posing problems | 2 | Participant C<br>Participant D | "There's always a challenge, always a problem to solve, there's not one way kids could cause problems trying to help them. It could be totally a thing I've never seen before. So it kind of keeps things a little more interesting, on my toes." (Participant D) |

| | | | |
|---|---|---|---|
| Thinking about thinking (Metacognition) | 2 | Participant H<br>Participant I | "And then once again that whole megacognition thinking about thinking, I'm always trying to imagine how they think." (Participant I) |
| Responding with wonderment and awe | 2 | Participant E<br>Participant I | "I need to keep wonderment and awe. All I need to always be going, "Oh my gosh, we can do this." (Participant I) |
| Thinking interdependently | 2 | Participant B<br>Participant J | "So my first one is thinking interdependently." (Participant J) |
| Applying past knowledge to new situations | 2 | Participant D<br>Participant E | "I guess this is a character trait that I have. I'm a very logical person." (Participant E) |
| Managing impulsivity | 1 | Participant A | "The grounds are hard. They're hard for me. Yeah, well, for some of my fifth graders but persisting through that managing impulsivity." (Participant A) |
| Gathering data through all senses | 0 | N/A | N/A |
| Listening with understanding and empathy | 0 | N/A | N/A |

**Research Question 3**

*What habits of mind do teachers of computer science name as important for learning?*

Similar to the results from question 2, study participants identified 14 of 16 habits of mind as being important for their students learning computer science. The most frequently mentioned habits of mind were persisting (n=6), applying past knowledge to new situations (n=4), thinking and communicating with clarity and precision (n=4), and taking responsible risks (n=4). The two habits of mind which were not mentioned were managing impulsivity and gathering data through all the senses. The numerous habits of mind listed as important for teachers of computer science further underscores the range of experiences and backgrounds of the study participants that have influenced what mental habits they believe are important. Table 5 showcases the habits of mind identified by the study participants, the number of participants that named the habit of mind as important, the specific participants that named the habit of mind, and an example of one participant's discussion of the habit of mind.


DOI: 10.4148/2637-4552.1164

Table 5. Important habits of mind for students

| Habits of Mind | Number of Participants | Participants | Example Evidence |
|---|---|---|---|
| Persisting | 6 | Participant A<br>Participant B<br>Participant E<br>Participant H<br>Participant I<br>Participant J | "In fact, one of the things that I say to them the first day we are together is to my intro kids is guess what, this is frustrating and there are days when you hate it and want to throw the computer, but when you solve the problem it feels really good. So be prepared to be frustrated every day. And know that, we just have to keep sticking to it until we figure it out." (Participant J) |
| Applying past knowledge to new situations | 4 | Participant A<br>Participant C<br>Participant F<br>Participant H | "Applying past knowledge to new situations, that's so important in coding because any problem that they're trying to address, an authentic problem is going to draw on what they have already created." (Participant F) |
| Thinking and communicating with clarity and precision | 4 | Participant C<br>Participant E<br>Participant H<br>Participant I | "Making them comment is very important...I've had kids stand up and talk about their code as well. They have to go through...they might have to stand up and say since my code is completely different then other groups, our team is going to talk about it." (Participant E) |
| Taking responsible risks | 4 | Participant B<br>Participant F<br>Participant G<br>Participant H | "I think taking responsible risks is really important. And I think that's...most of the kids. A lot of times the kids that have been straight A students in other classes often struggle the most when they come in. Just because they want a right answer. They want a right way to do it….but they're not so good at just trying things and |

| | | | |
|---|---|---|---|
| | | | they're really afraid of getting things wrong." (Participant G) |
| Questioning and posing problems | 3 | Participant A<br>Participant F<br>Participant G | "Problem solving is the number one indicator for me as a teacher, if a student will be successful in AP Principles when they're taking my coding one course. You know, they are going to encounter problems, challenges, and are they going to be the type of student who wants to ask a friend, who wants to ask me, who wants to Google it, and grapple with it, or are they a student who says, I'm not getting this after a couple minutes. I don't like this." (Participant F) |
| Thinking interdependently | 3 | Participant B<br>Participant C<br>Participant 4 | "I am hoping that they become more independent and more self-guided." (Participant C) |
| Thinking flexibly | 2 | Participant B<br>Participant D | "Flexibility, yeah, like that's part of the pair programming, you know, considering others." (Participant B) |
| Listening with understanding and empathy | 2 | Participant B<br>Participant D | "This is the idea of, you know, put yourself in the person who is going to use your and think about what they see, not what you're writing in the code because they don't get to see anything unless you put it on the screen. And so what is the message, the message that you want them to receive?" (Participant B) |
| Striving for accuracy | 2 | Participant B<br>Participant D | "We go over a debugging exercise and I give them, you know, like a problem that doesn't work, and [they] have to figure out why the errors are there." (Participant B) |
| Remaining open to continuous learning | 2 | Participant G<br>Participant I | "I want them to think about new technology. I want them to be surprised. There's nothing better |

| | | | |
|---|---|---|---|
| | | | than having a student yell out loud because they finally got something to work." (Participant I) |
| Finding humor | 2 | Participant C<br>Participant J | "Sixth grade boys love the party noise. You can't go wrong with that...they'll start making animations and they'll spin them around and they find humor in it. Absolutely. And they are so excited to share with their friends." (Participant C) |
| Thinking about thinking (Metacognition) | 1 | Participant I | "Thinking about thinking. That's a huge one, because they don't get that at all through their entire high school time do they actually get exposed to the idea of thinking about thinking, why they think the way they think, what their own preconceived biases are, what their natural tendency is towards how they learn, it's not something that they're exposed to so that's something we do, we do talk about that." (Participant I) |
| Responding with wonderment and awe | 1 | Participant I | "I encourage them to celebrate when we get something right and it happens a lot. And then it also drives the other one to get done whatever it was that the student got done and then they start asking them questions." (Participant I) |
| Creating, imagining, innovating | 1 | Participant J | "...because you have to think about things in a different way. And often times when you are coming at a problem you are having some of those problems I will have kids write down their ideas and then we talk about it. And they see that people are coming at that problem from a lot of different ways." (Participant J) |
| Managing | 0 | N/A | N/A |

| | | | |
|---|---|---|---|
| impulsivity | | | |
| Gathering data through all senses | 0 | N/A | N/A |

## Discussion

The findings of this study indicated that the computer science educators in this study used a variety of habits of mind to teach students computer science. Because students are taking computer science courses with varying levels of experience, educators rely on different teaching strategies to help to develop the habits of mind. For example, educators provided students with opportunities to make mistakes (and fix them) so that students are actively learning how to problem-solve, as well as identify problem areas in their code, which is a key component of becoming a successful programmer. Ismail et al. (2010) also found that active learning was essential for learning computer programming at a higher level. Most educators also described using various teaching strategies to address the challenges of teaching coding in their class. Whereas one educator may use pair programming as a strategy, they may also bring in experts to inspire and engage the students in the field of computer science. Sentance and Csizmadia (2016) found that it is common to use several teaching strategies when teaching computer science, and to adjust them with the changing needs of the students and the curriculum.

Teachers must also develop their own habits of mind to become successful computer science teachers. Most of the computer science educators in our study did not have a background in computer science, but were assigned to teach computer science due to their other abilities or interests. Therefore, many computer science educators learn coding while their students learn, which involves the habits of mind related to taking responsible risks, being open to continuous learning, and being able to find humor in situations. Several of our participants described how they were careful not to take themselves too seriously so that they could be open to learning new coding, and having students solve problems that they themselves may not be able to solve. In addition, most described making time to continue to develop their skills as the field of computer science was constantly changing and new material needed to be added frequently to their curriculum. Qian et al. (2018) found that novice computer science educators relied on professional development for developing content knowledge, but that experienced computer educators felt that they had less of a need for professional development. In addition, educators indicated that when they make mistakes or are unsure of how to do a specific skill, it inadvertently supports students who are also having the same experience when learning new content. Also, educators had unique habits of mind that were particular to their personalities, experiences, or courses. For example, one educator from an inner-city school indicated that humor was especially important to working with his population who may have experienced homelessness, violence or poverty, or work amongst classmates in these conditions.

The use, and emphasis, of different habits of mind aided educators in filling in gaps in their own knowledge. For example, Participant J admitted they were not comfortable with computer science and therefore, choosing to find humor in the situation increased their comfort level. However, educators who had more experience in computer science were more likely to emphasize striving for accuracy and thinking and communicating with clarity and precision. Yet, those same educators did not always have the same expectations for their students.

While teaching computer science requires specific habits of mind, students also need specific habits of mind to be successful in learning computer science. We asked all of the study participants which habits of mind they felt students needed and 14 of 16 habits of mind were mentioned. Study participants frequently named different habits of mind. However, the most common response was that students need to be able to persist, and not give up, through coding challenges. Students also needed to be able to communicate, take risks, and use past knowledge and experiences to continually improve their work. This often involved working through problems and fixing mistakes. Instructors can encourage students to have a growth mindset and provide opportunities for students to revise, solve problems and improve their work, collaboratively and through peer feedback. Study participants generally agreed that a computer science course was a safe place to make mistakes and often taught them to debug or work through errors.

## Conclusion

The findings from this study confirm the results Qian et al. (2018) found. The backgrounds of computer science teachers greatly impacted their pedagogy and the habits of mind used. In this study, participants came from various backgrounds. Few of them possessed any formal computer science training. These differences came out in their stories and are clearly shown in the wide range of habits of mind that were identified as important for both computer science teachers and computer science students. Instead of the identification of common habits of mind the participants believed to be important, the researchers instead discovered that the participants' varying backgrounds impacted how each teacher approached their courses as well as impacted what mental habits they valued. In other words, the teachers' beginning stories, and the events leading to the common climax (becoming current computer science teachers) led them to different endings when the focus is on the habits of mind necessary for success in teaching computer science and being a computer science student. For teacher preparation programs, this finding should encourage faculty to engage pre-service teachers and in-service teachers seeking professional development to add a computer science qualification area where they are. This means knowing the background and experience of future computer science educators, so they can use the mental habits they already possess in order to be successful teachers. Further, it also means encouraging future computer science teachers to embrace other mental habits they may not use as frequently.

However, due the range of habits of mind identified by the study participants, we question if all the habits of mind are needed, or if the lack of background in computer science and limited professional development as identified by Yadav et al. (2016) means that computer science teachers select habits of mind they are most comfortable with instead of fully understanding what the need, and what their students need, to be successful. Therefore, we recommend additional research on the habits of mind used by professional computer scientists to determine what mental habits should be fostered in the K-12 classroom.

## References


Carbone, A., & Kaasbøll, J. J. (1998, August). A survey of methods used to evaluate computer science teaching. In *Proceedings of the 6th annual conference on the teaching of computing and the 3rd annual conference on Integrating technology into computer science education: Changing the delivery of computer science education* (pp. 41-45).

Clandinin, D. J., & Connelly, F. M. (2000). *Narrative inquiry*. John Wiley & Sons.

Code.org. (2019). 33 states expand access to k-12 computer science education in 2019. *Medium.* Retrieved from https://codeorg.medium.com/32-states-expand-access-to-k-12-computer-science-education-in-2019-7d2357fe6f3d

Code.org, CSTA, & ECEP Alliance. (2020). 2020 State of Computer Science Education: Illuminating Disparities. Retrieved from https://advocacy.code.org/stateofcs

Council of Writing Program Administrators, National Council of Teachers of English, & National Writing Project. (2011). Framework for success in postsecondary writing. Retrieved from https://files.eric.ed.gov/fulltext/ED516360.pdf

Costa, A., & Kallick, B. (2008). *Learning and leading with habits of mind: 16 essential characteristics for success.* ASCD.

Kallick, B., Costa, A. L. (2009). *Habits of Mind across the curriculum: Practical and creative strategies for teachers*. ASCD.

KSDE. (2019). Kansas Computer Science Standards: Grades P-12. Retrieved from https://www.ksde.org/Portals/0/CSAS/Content%20Area%20(A-E)/Computer%20Science/Kansas%20Computer%20Science%20Model%20Standards%20with%20Description.pdf?ver=2019-04-23-165056-093

Costa, A. (n.d.). What are habits of mind. *Habits of Mind Institute*. Retrieved from https://www.habitsofmindinstitute.org/what-are-habits-of-mind/

Gal-Ezer, J., & Stephenson, C. (2014). A tale of two countries: Successes and challenges in K-12 computer science education in Israel and the United States. *ACM Transactions on Computing Education (TOCE)*, *14*(2), 1-18.

Gordon, M. (2011). Mathematical habits of mind: Promoting students' thoughtful considerations. *Journal of Curriculum Studies*, *43*(4), 457-469.

Guenaga, M., Eguíluz, A., Garaizar, P., & Gibaja, J. (2021). How do students develop computational thinking? Assessing early programmers in a maze-based online game. *Computer Science Education*, *31*(2), 259-289.

Ismail, M. N., Ngah, N. A., & Umar, I. N. (2010). Instructional strategy in the teaching of computer programming: a need assessment analyses. *TOJET: The Turkish Online Journal of Educational Technology*, *9*(2).

Kallick, B., & Costa, A. (2021). About us. *Habits of Mind Institute*. https://www.habitsofmindinstitute.org/about-us/

LeBlanc, C. (n.d.). *Tackling wicked problems*. Plymouth State University. https://wicked-problem.press.plymouth.edu/

Matsuura, R., Sword, S., Piecham, M. B., Stevens, G., & Cuoco, A. (2013). Mathematical habits of mind for teaching: Using language in algebra classrooms. *The Mathematics Enthusiast*, *10*(3), 735-776.

Nager, A., & Atkinson, R. D. (2016). The case for improving US computer science education. Available at SSRN 3066335.

Ni, L., McKlin, T., Hao, H., Baskin, J., Bohrer, J., & Tian, Y. (2021, August). Understanding Professional Identity of Computer Science Teachers: Design of the Computer Science Teacher Identity Survey. In *Proceedings of the 17th ACM Conference on International Computing Education Research* (pp. 281-293).

Rich, P. J., Browning, S. F., Perkins, M., Shoop, T., Yoshikawa, E., & Belikov, O. M. (2019). Coding in K-8: International trends in teaching Elementary/Primary computing. *TechTrends, 63*(3), 311-329.

Sentance, S., & Csizmadia, A. (2016). Computing in the curriculum: Challenges and strategies from a teacher's perspective. *Education and Information Technologies*, 1–27.

Schulte, C., Hornung, M., Sentance, S., Dagiene, V., Jevsikova, T., Thota, N., ... & Peters, A. K. (2012, November). Computer science at school/CS teacher education: Koli working-group report on CS at school. In *Proceedings of the 12th Koli Calling International Conference on Computing Education Research* (pp. 29-38).

Qian, Y., Hambrusch, S., Yadav, A., & Gretter, S. (2018). Who needs what: Recommendations for designing effective online professional development for computer science teachers. *Journal of Research on Technology in Education*, *50*(2), 164-181.

Umar, W. (2017). Constructing means ends analysis instruction to improve students' critical thinking ability and mathematical habits of mind dispositions. *International Journal of Education and Research*, *5*(2), 261-272.

Webb, M., Davis, N., Bell, T., Katz, Y. J., Reynolds, N., Chambers, D. P., & Syso, M. M. (2017). Computer science in K-12 school curricula of the 2lst century: Why, what and when? *Education and Information Technologies, 22*(2), 445-468.

Yadav, A., Gretter, S., Hambrusch, S., & Sands, P. (2016). Expanding computer science education in schools: Understanding teacher experiences and challenges. *Computer Science Education*, *26*(4), 235-254.

Yadav, A., Krist, C., Good, J., & Caeli, E. N. (2018). Computational thinking in elementary classrooms: measuring teacher understanding of computational ideas for teaching science. *Computer Science Education*, *28*(4), 371-400.