JTILT

Journal of Technology-Integrated Lessons and Teaching
(2025) Volume 4, Issue 1; DOI: 10.13001/jtilt.v4i1.9339

Integrating CT in Science Methods: Advancing Practice and Pedagogy

Ugur Kale[1] and Yuahnua Wang[2]
[1]Indiana University, [2]West Virginia University

Overview

Despite the importance of computational thinking (CT) as a problem-solving process (Wing, 2008) and the growing spread in teacher education (Yadav et al., 2017), existing initiatives for preservice teachers (PSTs) tend to focus on the computer science domain without making explicit connections to disciplinary classroom settings and promoting critical perspectives. As a cohesive unit, this learning representation aims to assist PSTs in integrating CT into their work as they design and implement science-focused lessons.

Centered around a contextual issue: accessing, growing, and sustaining food, this learning representation employs 2D and 3D block-based programming languages coupled with unplugged activities that demonstrate CT practices, processes, and concepts. PSTs' group designs, lesson modifications, and full lesson plans provide opportunities for assessment.

Topics: Computational Thinking, Science Education

Time: 18-20 hours

Materials

- Whiteboard and project connecting a computer
- Session 1 Slide Deck: Access Food
- Session 2 Slide Deck: Grow Food
- Session 3 Slide Deck: Sustain Food
- Group Lesson Design Template
- Modification Guideline
- Lesson Plan Template
- CT Lesson Implementation Rubric
- Garden-based learning supplies; plug trays, soil, cucumber seeds, LED grow light and heat mat
- Minecraft Education
- Scratch Account

Context-at-a-Glance

Setting
Undergraduate science methods courses in a teacher education preparation program in a largely rural state nested in the Appalachian region, U.S.

Modality
Face-to-face

Class Structure
The class meets twice a week. The CT component is addressed in three main sessions, each of which takes two weeks, offering knowledge, practice, and application. The room has six desks stationed mimicking a public-school science classroom layout.

Organizational Norms
Accredited by the Council for the Accreditation of Educator Preparation, the teacher education program supports preservice teachers' (PSTs) teaching knowledge via coursework and field experiences.

Learner Characteristics
Third-year undergraduate PSTs with limited CT knowledge. More than 93% were female and white. Slightly more than 25% are first-generation students.

Instructor Characteristics
The instructors have instructional design and science education expertise with good CT knowledge and familiarity with block-based programs.

Development Rationale
The learning representation addresses the existing CT initiatives' limited connection to disciplinary classrooms and critical perspective by providing opportunities for PSTs to relate CT to their practices.

Design Framework
TPACK, Computational Thinking Practices

Setup

The setup for the first and second sessions takes approximately 15 minutes while the third session, which involves the use of Minecraft Education additionally requires the installation of the software in advance. Although PSTs can install it on their laptops, a laboratory setting with computers already installed with Minecraft Education would ensure everyone has access to it timely in case of any technical difficulties with individual laptops. The instructors should review the slides and the templates prior to implementing the activities effectively. The schedule of the sessions can be adjusted according to the instructors' course timeline.

Standards

This learning representation aligns with the International Society for Technology in Education (ISTE) standard "*2.6.c Teach Computational and Design Thinking: Create learning opportunities that challenge students to use a design process and computational thinking to innovate and solve problems*" (ISTE, 2017).

Context and Setting

As a means to solve problems via concepts and techniques from the computer science field, CT has been considered an essential skill and explored in educational research (Tikva & Tambouris, 2021). The increasing interest and applications of emerging tools using Artificial Intelligence (AI) for education even further amplified CT's significance because a meaningful understanding of such systems, key to their effective use, also requires thinking computationally (Dohn et al., 2022; Hsu & Chen, 2024). Specifically, engaging in CT can entail making artifacts such as designing systems in situated environments reflecting the constructionist perspective (Harel & Papert, 1991). This process starts with a problem, which can be decomposed into smaller components, among which patterns can be identified helping to abstract a generic solution, followed by describing algorithms to automate and analyze the generalized solution (Barr et al., 2011; Grover & Pea, 2013; Wing, 2008). These elements closely mirror the problem-solving process including understanding and representing the problem, planning solutions, and executing solutions and self-regulating (Jonassen, 2000; Mayer & Wittrock, 2006; Polya, 1957).

Despite CT's importance as a problem-solving process (Wing, 2008) and its growing spread in teacher education (e.g., Adler et al., 2023; Yadav et al., 2017), two main issues remain. First, the current CT-focused initiatives solely focus on the computer science domain without explicit connections to disciplinary classroom settings (Yadav, et al., 2018). This is problematic because as the key emphasis of the Technological Pedagogical and Content Knowledge (TPACK) framework, a successful integration of technological innovations in teaching requires not only the knowledge of the innovation but also the knowledge of how to use it in particular subject-matters (Mishra & Koehler, 2006). As such, without opportunities to understand and practice CT in relation to their teaching and particular content, PSTs' future implementations with CTs will likely be impeded (Kale et al., 2018; Rich et al., 2019).

There are efforts to connect CT to various subjects (Barr & Stephenson, 2011). To clarify the meaning of CT specifics to mathematics and science practices, for instance, Weintrop et al. (2016) developed a taxonomy with four main categories that can elicit CT processes via various practices:

1. Data Practices (e.g., collect, create, analyze)
2. Modeling and Simulation Practices (e.g., use, assess, design, and construct)
3. Computational Problem Solving Practices (e.g., abstracting and debugging)
4. Systems Thinking Practices (e.g., understanding relationships within a system)

Although the practices look distinct, they are not mutually exclusive, and the last two categories reflect rather a problem-solving process that can be engaged simultaneously with the practices. For instance, learners may analyze data (Data Practices) or construct models (Modeling and Simulation Practices) as they try solutions to a given problem (Computational Problem-Solving Practices) that may involve understanding the relationship among multiple factors (Systems Thinking Practices). Thus, differentiating (1) Data and (2) Modeling and Simulation as CT practices from (3) Computational Problem Solving and (4) Systems Thinking as CT processes, which involve decomposition, pattern recognition, abstraction, algorithms & automation, and analysis, will be a more accurate approach (see Figure 1 for a modified taxonomy).

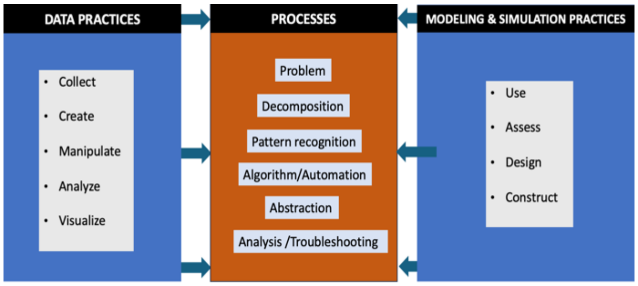


Figure 1. CT practices supporting CT processes.

Second, CT has been viewed mainly as a cognitive outcome, but a rather critical perspective of CT has emphasized self-expression, connection, and questioning (Brennan & Resnick, 2012) and been recently considered as a potential way to engage learners in the social, economic, political challenges of the world that directly impact them (Kafai et al., 2019; Tissenbaum et al., 2019). However, only a few examples exist that focus on such contextual issues via CT such as political resistance in the form of Latino students' storytelling (Aghasaleh et al., 2019) and the design of an environmental awareness mobile app (Tissenbaum et al., 2019).

To address these two issues and to help PSTs gain competence with and increase their interest in CT, a critical perspective of CT was infused into a science methods course offered in an undergraduate elementary teacher education program. Approximately seventy-five PSTs in their third year in the teacher education program enrolled in the course and more than 93% were female and white. According to the West Virginia University Office of Data and Analytics (2024), more than a quarter of undergraduate students enrolled in the college identified themselves as first-generation students. Further, the PSTs' responses to a question in a pre-survey instrument indicated they were between slightly aware and moderately aware of the social, economic, or political challenges in the state. PSTs' responses to this pre-survey instrument (see Kale et al., 2024; Kale & Wang 2024) also showed their varying level of competence in CT, perception of CT's value, interest in CT, and use of CT in teaching. On average, PSTs either agreed or strongly agreed that CT can be relevant to their teaching practices. They also found CT interesting but did not feel competent to integrate it in teaching. A majority reported limited frequency of CT usage (e.g., a few times a semester). The course focused on in-class activities helping PSTs unpack Next Generation Science Standards (NGSS), assess student thinking, modify an existing science lesson and teach it in their placement schools, and reflect on their teaching. PSTs spend around 10 hours per week in placement classrooms during the semester.

As part of a federally funded project, this learning representation was developed as a CT unit by the authors and various experts such as elementary school teachers and technology integration specialists. The unit centered around a contextual issue - accessing, growing, and sustaining food. For context, the state where the program is located is largely rural, nestled in the Appalachian Mountains in the U.S., and faces an abundance of challenges. Besides the poor health status and low median income, the area this lesson took place is ranked one of the highest in the nation regarding the uncertain availability of nutritionally adequate and safe food at the household level (Gundersen et al., 2020). And, state residents with limited food access tend to be located in certain locations, often labeled as "food deserts" where temporal, spatial, and socio-economic factors act as barriers (Wilson et al., 2017).

Focusing on the food access issue via CT, three sessions were provided, and a Garden-based Learning (GBL) experiment was conducted. The sessions involved two classroom days, each of which lasted 75 minutes, and took pace in a STEM lab. Originally designed to mimic the layout of a science classroom, it has six desks encouraging group work. One session, focusing on the use of Minecraft Education, as described in the next section, was completed in a computer lab where the software was installed onto computers in case the program does not work in the PSTs' laptops.

Learning Representation

During this lesson, italic text identifies prompts for the learners.

Session 1: Access Food

This session is a total of six hours that was spread across four in-class days (1.5 hours each) and occurred during Weeks 2-3 of the semester.

INTRODUCTION (1.5 HOUR)

To start this learning representation, the instructor should first demonstrate to PSTs how the CT approach can be used to understand and deal with *Food Access*. The Session 1 Slide Deck: Access Food (PPT; slides 2-64) includes a detailed explanation of the steps covered for this first session.

In Session 1, the instructor started with a question, "*What's the sum of numbers between 1 and 200?*" and asked the PSTs to think about how to solve it without using any calculators. This was just a brief example about demonstrating the main processes of CT, namely, decomposing the problem (e.g., 1, 2, 3...... 198, 199, 200), identifying the patterns (e.g., 1+200 = 201, 2+199 = 201, 3+198 = 201) and solving the given problem (e.g., Solution = 201*100 pairs), abstracting a generic solution (e.g., for a given number *N*, the Solution = N+1*(N/2)), and developing the algorithm (e.g., outlining steps for a computer program to follow), and analyzing its effectiveness (e.g., testing with different numbers).

The instructor then introduced CT processes as a means to solve problems and highlighted how programming tools may provide opportunities to practice these processes. Two unplugged activities (see Figure 2 and Figure 3) were used to help the PSTs practice the development of algorithms without any technology tools.

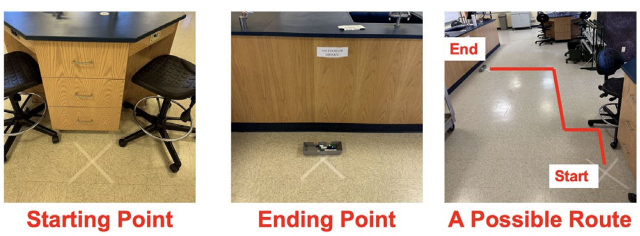


Figure 2. Activity 1 (see Slide 24 in Session 1 Slide Deck: Access Food (PPT)).

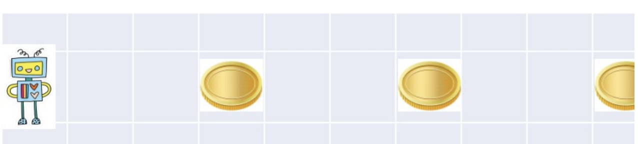


Figure 3. Activity 1 (See slide 26 in Session 1 Slide Deck: Access Food (PPT)).

The first unplugged activity required two volunteers: one PST to act as a robot tasked with moving a marker from one location of the room to another, and another PST to provide directions to the "robot". The second unplugged activity required everyone in the class to come up with the steps a robot, displayed on board, needs to follow in order to collect the displayed coins. After a discussion of what CT processes are addressed in these two activities, another activity followed the same format but rather focused on a scientific concept: the pumpkin growth cycle where PSTs worked on correctly sequencing the given images of the cycle. A few examples of scientific topics such as the state of matter and the water cycle were also discussed regarding the specific CT processes elicited.

To help PSTs further engage in their understanding of the CT processes, the instructor asked them to work in groups to draw an equilateral triangle on a paper and list the steps followed. In each group, one person controlled the pen while the others provided directions to follow. Having shared their drawings and steps (e.g., number of sides to draw and angles to turn), the groups were prompted to draw a square and again share their drawings and steps. This was followed by another prompt where PSTs were asked to recognize the relationship between the kind of shape and its properties (see Figure 4). The PSTs also discussed CT processes elicited in these examples.

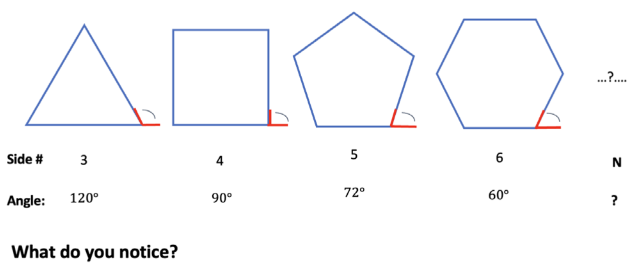


Figure 4. Properties of chapes compared (See Slide 50 in Session 1 Slide Deck: Access Food (PPT)).

After the discussion, the PSTs were prompted to draw the shapes in Scratch for the next fifteen minutes. Prior to the class, PSTs were asked to create accounts in Scratch, get familiar with the interface, and practice some of the codes so they could be ready for this part of the activity. For this, they were guided to use the pen blocks to come up with the algorithm for drawing a triangle first then a polygon with any number of sides given by the user, which necessitates the abstraction process (see Figure 4). PSTs were then prompted to identify the functions of various coding blocks that they use.

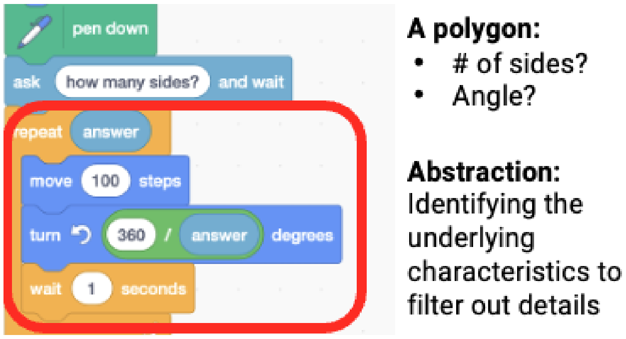


Figure 5. Algorithm to draw a polygon (See Slide 60 in Session 1 Slide Deck: Access Food (PPT)).

Next, the instructor showed another example idea that can be developed in Scratch. This focused on another science concept: energy, and the session ended with a discussion about how coding and programming may be connected to the NGSS (n.d.), which they started to review in the previous week.

Demonstration (1.5 hour)

This part of the activity took place on the second classroom day of the same week (e.g., Thursday of a Tuesday/Thursday class). The Session 1 Slide Deck: Access Food (PPT; slides 65-88) includes a detailed explanation of the steps covered for Day 2. The instructor started by reminding the PSTs of the previous activity of drawing shapes and shows a completed program in Scratch. Then, two example Scratch projects: one simulating the relation between voltage, current, and resistance, and the other one simulating the population of foxes and rabbits changing over time were presented, followed by the instructor discussing the role of simulation in teaching and learning science concepts (see Figure 5).

Figure 6. Simulations on Ohm's law and the dynamics of two biological systems (see Slide 69 & 71 in Session 1 Slide Deck: Access Food (PPT)).

PSTs were also prompted to consider any contextual issues that CT may be used for besides the cognitive learning benefits that such simulation examples can offer. To facilitate the discussion, the instructor displayed another simulation that they had previously developed, which shows how modifying factors such as dispersant amount and exposure to sunlight impact how fast an oil spill can be cleaned (see Figure 6).

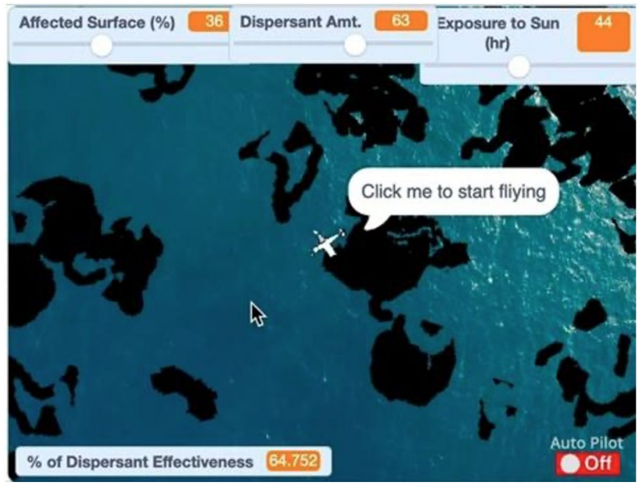


Figure 7. Oil Spill Simulation (See slide 74 in Session 1 Slide Deck: Access Food (PPT)).

Following the simulation, the instructor provided an overview of contextual issues that were more specific to the state focusing on food access. Another Scratch project that the instructor developed previously, which simulates how food access varies depending on a given county, was presented. The PSTs were asked to run the simulation over a certain amount of time, generate data in Excel, and try to identify the equation used in the simulation for the food access index (see Figure 7).

Grocery	Convenience	Food index
1	5	125
4	10	450
6	2	610
3	5	325
8	6	830
19	18	1990
10	3	1015
13	10	1350
19	12	1960

Figure 8. Simulation on food access, and generated data entered in Excel Slides (See Slide 79 in Slide Deck #1).

After 20-30 minutes, the PSTs shared, discussed, and compared their equations and were asked to consider other factors (e.g., family income) impacting food access. In the end, they were tasked with focusing on any of such factor and coding them in the original simulation before the next classroom. They were asked to use commands engaging in the first three basic CT concepts (e.g., sequences, loops, and events) adapted from Brennan and Resnick's (2012) work on the assessment of CT.

Practice & Application (3 hour)

After the introduction and demonstration in the previous week, the next class focused on practicing and modifying the original Scratch simulation assigned. The Session 1 Slide Deck: Access Food (PPT; slides 89-94) includes a detailed explanation of the steps covered for this practice day. While the instructor modeled, the PSTs were encouraged to complete the coding tasks on their own first, ask questions, and troubleshoot difficulties encountered by others. They continued modifying each project to reflect the new factors to take into account, and remixed other PSTs' projects shared in an online discussion forum.

In the next class day (see slides 95-97 in Session 1 Slide Deck: Access Food (PPT)), having practiced and completed modifying the simulation individually, the PSTs worked in groups of three to start describing a learning activity that incorporates CT in teaching a science concept of their choice. The groupings were based on the grade level of their field experience placement classrooms. They were given a template (see Group Lesson Design Template) to complete their descriptions which would entail determining the contextual issues to be addressed, developing a timeline of planned instructional events, and identifying the CT practices, processes, and concepts that their activity aims to promote. A list and description of CT components were included in the very template for their reference. The PSTs were to submit their group learning activity description to an online discussion forum in a week.

After the group work, the PSTs were given another week to make one modification to an existing science lesson that they acquired from their placement classrooms or a published resource. They were prompted to use their experiences with the sessions and group learning activity so far to modify such an existing plan with a focus of promoting students' CT (see Modification Guideline). They described the existing lesson plan and the modifications as well as responding to reflection questions (e.g., *what are some CT processes that the students would be engaged in this lesson?*). Their responses informed their full lesson plans that they develop throughout the semester (see Lesson Plan Template).

Session 2: Grow Food

This session is a total of six hours that was spread across four in-class days (1.5 hours each) and occurred during Weeks 6-7 of the semester.

Introduction (1.5 hour)

In Week 6 of the semester, the instructor and a guest speaker, a former science educator who also has experience with Garden-Based Learning (GBL) demonstrated to the PSTs how the CT approach can be used to understand and deal with *Grow Food*, one of the means to address the food access issue. Community gardens and active participation in gardening, for instance, have been considered one of the ways to increase local food production (Furness & Gallaher, 2018). As such, engaging the PSTs in basic gardening techniques and employing inquiry skills would expand the concept of food access covered in the first session.

To prepare for the sixth week and to generate some data to analyze, the instructor and the PSTs launched a simple experiment two weeks prior to this classroom. In Week 4 of the semester, the PSTs were guided to work in four groups to sow cucumber seeds in a hydrated potting mix inside plug trays, each of which has 36 cells. A guiding question for the PSTs as well as for an inquiry that they can use in their future classrooms could be "*We cannot harvest cucumbers in this region until early Summer. Can we germinate the seeds earlier in our classroom? What will be the most ideal condition to germinate them when considering the heat, light, and kinds of seeds*?

To examine the impact of heat (no heat-NH, heat-H) and light (normal light-NL, extra light-EL) on germination rates, each group's tray had a different set up representing one of 4 possible variations (see Figure 9):

- Group 1: The tray had no heat mat and was exposed to normal daylight during the day (NH and NL).
- Group 2: The tray had no heat mat (NH) but was exposed an extra light source (EL): an LED grow light placed over it.
- Group 3: The tray was placed on a heat mat (H) and was exposed to normal daylight (NL).

- Group 4: The tray was placed on a heat mat (H) and was exposed an extra light source (EL): an LED grow light placed over it.

To determine the germination rates, groups observed the trays every day until the sixth week and counted the total number of seeds that germinated each day.

Figure 9. Observed outcomes of the experiment.

On the first classroom day of the sixth week, the guest speaker introduced the tenets of GBL and an example project that is being implemented in a local public elementary school. Prior to the session day, he already shared with the PSTs relevant readings and questions about their perspectives on and experiences with GBL. In the classroom, he highlighted specific student artifacts focusing on the germination process in relation to students' science concepts and mathematical understanding and the science and engineering practices of the NGSS. He also directed the PSTs to further resources such as books and guidance about GBL (see Figure 10). The Session 2 Slide Deck: Grow Food (PPT; slides 2-31) includes a detailed explanation of the steps covered.

Figure 10. Guest speaker presenting GBL resources.

DEMONSTRATION (1.5 HOUR)

After conducting the cucumber seed experiments and learning about GBL, the PSTs were ready to analyze their collected data and develop a simulation. In this second classroom day of the sixth week, the instructor reminded the PSTs of the conditions set for their experiments and prompted them to compare the observed outcomes with the focus of identifying any patterns associated with each group. To help facilitate their thinking, the instructor helped them enter the total # of seeds that germinated each day for their condition in a Google Sheet. The PSTs were also guided to develop scatter charts to compare the observed outcomes of the conditions and discuss the relationship between the day and the number of seeds (See Figure 11).

Group 1 (NLNH) heatoff =0 Light off=0		Group 3 (YLNH) Heat off=0 Light on=1	
Days	Total seeds	Days	Total seeds
1	0	1	0
2	0	2	2
3	0	3	2
4	0	4	3
5	1	5	3
6	1	6	4
7	1	7	4
8	2	8	5
9	2	9	5
10	2	10	5

Group 3 (NLYH) Heat On =1 Light off=0		Group 4 (YLYH) Heat On =1 Light on=1	
Days	Total Seeds	Days	Total seeds
1	0	1	1
2	0	2	2
3	2	3	3
4	2	4	3
5	3	5	4
6	3	6	5
7	3	7	6
8	4	8	6
9	4	9	6
10	4	10	6

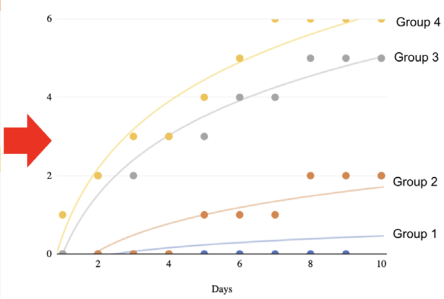


Figure 11. Observed data keyed in and visualized.

Toward the end, the instructor displayed a simplified simulation that allowed users to control multiple conditions and see the effects on the germination process (See Figure 12).

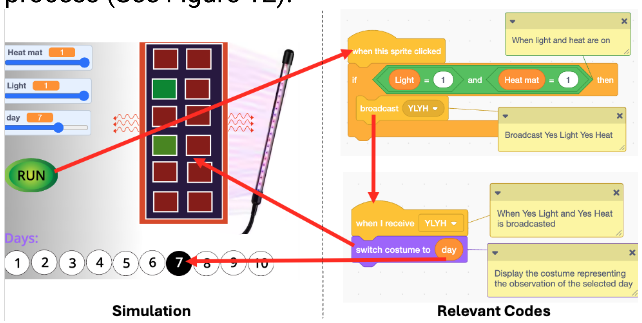


Figure 12. Sample simulation and its relevant codes.

The PSTs were then provided with a link to another version of the same simulation that had partially completed codes. The PSTs were asked to work in groups till the end of the class to finish the coding so that the program would simulate the data observed in their assigned conditions. The instructor supported the PSTs during this time by discussing their codes.

For homework, to be completed before the next class session, the PSTs were tasked with individually developing a new simulation of their choice in Scratch with the observed cucumber seed data. Given their developing experiences with Scratch and the potential complexity of the created simulations, they were provided specific commands to use engaging in both basic and advance CT concepts (e.g., sequences, loops, events, conditionals, and data) adapted from Brennan and Resnick's (2012) work. The Session 2 Slide Deck: Grow Food (PPT; slides 32-43) includes a detailed explanation of the steps covered in this demonstration class.

PRACTICE & APPLICATION (3 HOUR)

In the following week, the session followed the same format as the Session 1: Access Food practice and application. This practice and application occurred during Week 7 of the semester, encompassing both in-class meeting days. To start, the PSTs brought their individual Scratch simulations to class (the previous week's homework). These simulations were reviewed and modified in both peer-groups and individual instructor review. The instructor again modeled and encouraged them to work on first updating the code of their simulations on their own with the peer and instructor feedback and troubleshooting difficulties encountered by peers.

On the second classroom day of this week, PSTs started working with the Group Lesson Design Template in their previous groups to create a new learning activity incorporating CT in their field experiences the following week. Based on their newly gained understanding from the cucumber seed experiment and Scratch simulations, PSTs used another week to individually modify the existing lesson plan (that they chose last time) toward supporting students' CT. The Session 2 Slide Deck: Grow Food (PPT; slides 44-52) includes a detailed explanation of these steps.

SESSION 3: SUSTAIN FOOD

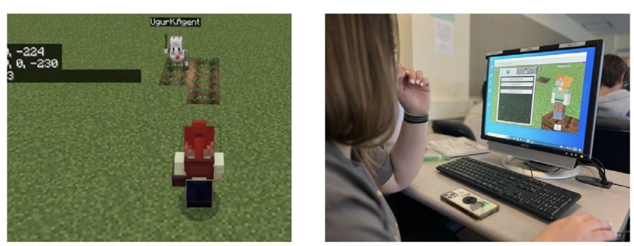
This session is a total of six hours that was spread across four in-class days (1.5 hours each) and occurred during Weeks 10-11 of the semester.

INTRODUCTION (1.5 HOUR)

In the 10th week of the semester, the instructor demonstrated to the PSTs how the CT approach could be used for sustaining food, another key factor for local and healthy food access. Small-scale farming is critical to local and healthy food access (Blackwell, 2016) yet sustainability of food from such sources can be problematic. To make farming more sustainable, agricultural robots have been explored to automate various dull tasks such as picking, harvesting, weeding, spraying, or monitoring for farmers (Dao, 2020; Robotics Online Marketing Team, 2017). While experimenting with such robots may not be feasible due to their high cost, simple and accessible programming platforms like Minecraft (a 3D virtual space where users develop structures via manipulating blocks) have been explored to model a variety of robotics tasks (Aluru et al., 2015). Thus, engaging PSTs in the use of such tools to practice automating relevant tasks would further their understanding of food access and growth covered in the first two sessions. The guiding question for the PSTs, as well as an inquiry they can use in their future classrooms, was: "*A large number of cucumber seeds need be planted in a family-owned farm but it takes a lot of work for the family to prepare the soil and plant the seeds in a short amount of time. Can we automate these tasks such that they are more efficiently completed for a specified planting area layout as well as for any given planting layout*?

To prepare for this week, the PSTs were instructed to download the Minecraft education edition onto their computers and practice navigating some of the existing worlds a week prior to the session. On the first classroom day of the tenth week, the instructor started by presenting information on the agricultural practice and trends in the state and displayed the example uses of robots in agriculture as a means to deal with the sustainability of food in the context. The instructor then introduced and discussed the use of Minecraft (education edition) as an efficient and effective platform to test robotic tasks without using robots in the real world (see Session 3 Slide Deck: Sustain Food (PPT; slides 1-14).

Next, the instructor presented a previously developed Minecraft world (see Figure 13), which simulated the automation of farming tasks (e.g., tilling, planting) via the agent- a robotic-looking character executing the given commands. During this time, he prompted the PSTs to discuss the possible algorithm used, and shared the URL of the world, which the PSTs used to join and explore it for the next 10-15 minutes.



Minecraft World Shared   Minecraft World Explored

Figure 13. Existing Minecraft world shared and explored.

In the remaining part of the class, the PSTs practiced using the Minecraft Education interface such as moving the player around, placing or destroying blocks, opening the code builder window, using commands to code the agent, a character simulating the robot to move it around, make it place and destroy blocks, or teleport it to specific coordinates. The Session 3 Slide Deck: Sustain Food (PPT; slides 2-29) includes a detailed explanation of the steps covered for Day 1.

Demonstration (1.5 hour)

On the second classroom day of week 10, the PSTs were asked to examine the original algorithm used in the previous demonstration, practice modifying it, and observe its impact on the agent's task completion. This time, the instructor shared with the PSTs the URL of the coding file, which the PSTs would use in Minecraft Education to be able to review the codes (see Figure 14).

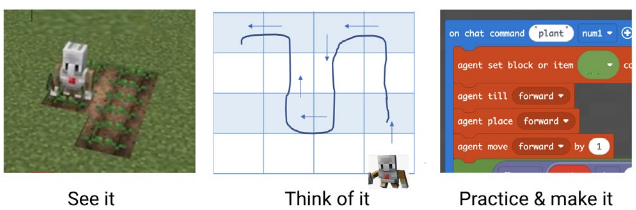


Figure 14. Code file shared, discussed, and modified.

Having explored the codes, the PSTs were asked to work in groups to create new commands that enables the agent to perform the same tasks but in a new planting area with a different size (e.g., 9X9). The instructor monitored the PSTs progress and helped them modify the original codes for the new planting area. At the end of the session, the PSTs were tasked with developing a new world with new commands to simulate the automation of any task prior to the next classroom. They were encouraged to think about their own lessons and accordingly create a world (e.g., biomes) with relevant structures (e.g., buildings, lakes, roller coasters, waterslides) and animals (e.g., cows, wolves, fish). Given their limited experiences with the interface of Minecraft's coding platform (Makecode), they were asked to use commands engaging mostly in the basic CT concepts (e.g., sequences, loops, events). The Session 3 Slide Deck: Sustain Food (PPT; slides 30-47) includes a detailed explanation of the steps covered in Day 2.

Practice & Application (3 hour)

In Week 11 of the semester, the students engaged in the practice and application portion of Session 3, following the same format of the practice and application portions for Session 1: Access Food and Session 2: Grow Food. In the first day of class in Week 11, the PSTs continued working on their Minecraft simulation and modifying the code file while the instructor continued to model and encourage them to troubleshoot the difficulties encountered by their peers.

On the second class day of Week 11, the PSTs got into their groups, using the Group Lesson Design Template, to create a new learning activity incorporating CT for their field experience the following week. Based on their newly gained understanding from the demonstrations in-class and the group work, the PSTs used another week to individually work on modifying the existing lesson plan (that they already chose last time) toward supporting students' CT. The Session 3 Slide Deck: Sustain Food (PPT; slides 48-56) includes a detailed explanation of the steps covered in these practice and application classes.

Assessment Opportunities

The assessment related to the three sessions had two main components. The first component provided the PSTs with an opportunity to reflect on CT right after each session. As described in the practice and application sections, the PSTs worked in groups to design a learning activity incorporating CT into teaching science concepts. The Group Lesson Design Template guided the PSTs to describe the contextual issues to be addressed and the sequence of the instruction to take place as well as identifying various CT components (see Table 1).

Table 1. CT Components Reflected

CT Components	Specific Inclusion
Data Practices	• Collecting data • Creating data • Manipulating data • Analyzing data • Visualizing data

CT Components	Specific Inclusion
Simulation and Modeling Practices	• Using computational models to understand concepts/find/test solutions • Assessing computational models • Designing computational models • Constructing computational models
Processes	• Confrontation • Decomposition • Pattern recognition • Abstraction • Algorithm/Automation • Test/Debug
Concepts	• Sequences • Loops • Events • Parallelism • Conditionals • Operators • Data

As also described earlier, the PSTs worked individually, after their group design, making one modification to an existing science lesson that they acquired from their placement classrooms or a published resource. The Modification Guideline helped their reflection where they described the existing lesson, the modifications they made, and how the modifications will enable their students' CT.

The second component of the assessment was embedded in the PSTs' individual lessons. Having practiced with CT tools in three sessions (Scratch, Excel/Google Sheets, and Minecraft), designed group lessons on CT, and modified existing lessons, the PSTs individually taught full lessons at their placement classrooms incorporating CT. The Lesson Plan Template was shared with them at the beginning of the semester while its elements such as objectives, assessment, and instructional strategies were reviewed throughout the semester to help guide their lesson development. A section toward the end of the Lesson Plan Template focuses on the enactment of teaching where the PSTs respond to questions such as:

- *Based on your implemented lesson, what specific CT practices and processes did you notice students were engaged in? What evidence do you have that indicates students practice computational thinking?*
- *How do you think the modifications better supported your students' conceptual understanding of the lesson focus? What evidence do you have that supports your thinking (i.e. what did you see or hear students say and do that supports why you think the modifications were effective or not towards supporting students' thinking and reasoning)?*

A potential rubric could also further guide PSTs' teaching of CT-infused lessons (see CT Lesson Implementation Rubric).

Critical Reflection

The learning representation was implemented with two cohorts of PSTs in Spring 2023 and Spring 2024. The main goal of infusing CT into the science methods course was to help PSTs gain competence with and increase interest in CT so that they can incorporate it into their science instructional practices.

In each semester, we collected both pre and post surveys as well as conducted interviews focusing on PSTs' competence with, motivation for, and use of CT. As part of the larger research project, we reported our detailed analysis and results in relevant research (Kale et al., 2024, Kale & Wang, 2024). In this section, we provide our reflections on the outcomes for both cohorts, describing major findings along with modifications to the curriculum.

For the first cohort, we observed that the PSTs' competence with CT in teaching science as measured via the survey items at the end of the semester was significantly higher than those at the beginning of the semester. Most PSTs also reported that their understanding of CT improved after completing the course, which had a positive effect on their ability to use it.

However, the PSTs' interest in CT and perception of CT's utility value (e.g., how relevant they consider CT to teaching) decreased over time while no change was observed regarding the frequency of their CT usage. Congruent with our earlier research (Kale & Akcaoglu, 2017), these findings suggest that the initial interest and perceived value of a technological innovation can diminish if users encounter difficulties with its use. Interview responses emphasized that the use of coding and the complexity of simulations posed challenges for the PSTs, which may have led to a decrease in the initial novelty of CT. This was also reflected in their midterm course evaluations, such as the comment: "*I don't enjoy the coding assignments because they are too complex and confusing for me, let alone my elementary students".* The perceived limited application of CT in placement classrooms appeared to be another factor. One PST noted in their midterm evaluation "*I do not understand why we have a whole class about computational thinking when it is not used in our placement classrooms*."

Based on the findings from the first cohort, we made two main revisions to the curriculum. The first one involved simplifying the coding tasks. As part of Session 1: Grow Food, the PSTs were asked to create a simulation in Scratch to display the number of seeds germinating. The complexity of such coding was amplified especially when the PSTs were introduced to the trendlines and generation of the equation formula in Excel, which would be coded in Scratch to produce the total number of seeds that germinate on any given day. Although using the formula can generate results in the simulation beyond the observation timeline, the PSTs considered it difficult to comprehend. As such, we simplified the coding task by removing the formula generation. Rather, the task can focus on identifying the number of seeds observed in the week only and creating simple conditionals for each day in Scratch (e.g., if Day 1 is selected, then display 2 seeds being germinated), which is much easier for the PSTs to practice. An alternative approach may be to provide pre-training opportunities on the use of Excel for trendline and equation. However, the existing course timeline and other assignments did not allow for such an additional activity.

The second main revision was the use of open-ended personalized coding tasks. Toward the end of the demonstration sections of each session, the PSTs were tasked with completing close-ended tasks such as working on an existing simulation (e.g., coding additional factors in the Food Access simulation). While these tasks allowed practicing many CT concepts, the PSTs, without planning and choosing their own topic, may not have developed the ownership on their coding projects, which may be essential to their motivation (Gonzalez-Maldonado et al., 2022). As such, in addition to closed-ended tasks, we recommend incorporating open-ended, personalized ones, such as selecting, planning, designing, and developing their own projects similar to the one we used in Session 3: Sustain Food where the PSTs are asked to design and develop their own Minecraft worlds.

For the second cohort, we found a similar outcome regarding the competence. The PSTs felt significantly more competent with CT at the end of the semester than at the beginning of the semester. Again, this observation reflects the effectiveness of the activities on PSTs' developing competence. However, unlike the first cohort, the second cohort of PSTs (Spring 2024) were able to maintain their interest and perception of CT's value while a slight increase in their usage was also observed toward the end of the semester. Given the lowered motivations observed with the first cohort, these second cohort observations are promising, which speak to the positive impact of the changes made to the curriculum.

In addition, two potential changes can further help the PSTs recognize the relevance of CT (utility value). The first one requires stronger connections between CT and NGSS (n.d.). This could simply involve activities that more frequently explore the connections between the dimensions of NGSS, such as Science and Engineering Practices and Crosscutting Concepts, and both CT processes and practices (See slide 84 in the Session 1 Slide Deck: Access Food (PPT)). The second one involves exemplifying CT in grade-specific classrooms. Observable pedagogies via existing classroom videos (e.g., Brush et al., 2009; Luna & Sherin, 2017) can be a good solution here though finding or developing such cases in timely manners may not be feasible. PSTs videoing their teaching practices via their mobile phones and securely sharing it with their peers may provide an alternative. While not all of these cases would represent exemplary practices, the challenges faced, the strategies used to overcome them, and the successes seen from peers' perspectives could serve as valuable examples to demonstrate CT's application, thereby enhancing its perceived relevance to teaching.

References

Aluru, K.C., Tellex, S., Oberlin, J.G., & MacGlashan, J. (2015). Minecraft as an experimental world for AI in robotics. *AAAI Fall Symposia*. https://aaai.org/papers/11725-11725-minecraft-as-an-experimental-world-for-ai-in-robotics/

Adler, R. F., Hibdon, J., Kim, H., Mayle, S., Pines, B., & Srinivas, S. (2023). Assessing computational thinking across a STEM curriculum for pre-service teachers. *Education and Information Technologies, 28*, 8051-8073. https://doi.org/10.1007/s10639-022-11508-4

Aghasaleh, R., Enderle, P., & Puvirajah, A. (2019). From computational thinking to political resistance. *Journal for Activist Science and Technology Education, 10*(1), 29-44. https://doi.org/10.33137/jaste.v10i1.32915

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community? *ACM Inroads, 2*(1), 48-54. https://doi.org/10.1145/1929887.1929905

Barr, D., Harrison, J., & Conery, L. (2011). Computational thinking: A digital age skill for everyone. *Learning & Leading with Technology, 38*(6), 20-23.

Blackwell, A. (2016, May 12). *Best practices for creating a sustainable and equitable food system in the United States.* Center for American Progress. https://www.americanprogress.org/issues/poverty/reports/2016/05/12/137306/best-practices-for-creating-a-sustainable-and-equitable-food-system-in-the-united-states

Brennan, K., & Resnick, M. (2012, April). *New frameworks for studying and assessing the development of computational thinking.* In Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada. https://scratched.gse.harvard.edu/ct/files/AERA2012.pdf

Dao, E. (2020, May 18). This professor wanted to make farming more sustainable, so he started a company to build a fleet of smart robot. *TerraMapp.* https://terra-mepp.illinois.edu/news/professor-wanted-make-farming-more-sustainable-so-he-started-company-build-fleet-smart-robots

Dohn, N. B., Kafai, Y., Mørch, A., & Ragni, M. (2022). Survey: Artificial intelligence, computational thinking and learning. *KI-Kunstliche Intelligenz, 36*(1), 5-16. https://doi.org/10.1007/s13218-021-00751-5

Gonzalez-Maldonado, D., Pugnali, A., Tsan, J., Eatinger, D., Franklin, D., & Weintrop, D. (2022, August). *Investigating the use of planning sheets in young learners' open-ended scratch projects.* In Proceedings of the 2022 ACM Conference on International Computing Education Research-Volume 1 (pp. 247-263). https://doi.org/10.1145/3501385.3543972

Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher, 42*(1), 38-43. https://doi.org/10.3102/0013189X12463051

Gundersen, C., M. Hake, A. Dewey, E. Engelhard (2020). The impact of the coronavirus on food insecurity v1 [Data file and FAQ]. Available from Feeding America: research@feedingamerica.org.

Harel, I., & Papert, S. (1991). *Constructionism*. Westport, CT, US: Ablex Publishing.

Hsu, T. C., & Chen, M. S. (2024). Effects of students using different learning approaches for learning computational thinking and AI applications. *Education and Information Technologies, 30*, 1-23. https://doi.org/10.1007/s10639-024-13116-w

International Society for Technology in Education. (2017). *ISTE standards: Educators*. Retrieved May 14, 2025, from https://www.iste.org/standards/iste-standards-for-teachers

Jonassen, D. H. (2000). Toward a design theory of problem solving. *Educational Technology Research and Development, 48*(4), 63-85. https://doi.org/10.1007/bf02300500

Kafai, Y., Proctor, C., & Lui, D. (2019, July). *From theory bias to theory dialogue: Embracing cognitive, situated, and critical framings of computational thinking in k-12 cs education*. In

Proceedings of the 2019 ACM Conference on International Computing Education Research (pp. 101-109). https://doi.org/10.1145/3291279.3339400

Kale U., & Akcaoglu, M. (2017). The role of relevance in future teachers' utility value and interest toward technology. *Educational Technology Research and Development. 66*(2), 283-311. https://doi.org/10.1007/s11423-017-9547-9

Kale, U., Akcaoglu, M., Cullen, T., Goh, D., Devine, L., Calvert, N., & Grise, K. (2018) Computational what? Relating computational thinking to teaching. *TechTrends*, *62*(6) 574-58. https://doi.org/10.1007/s11528-018-0290-9

Kale, U., Wang Y., Aldebyan, A., Mann T., & O'Brian, N. (2024, April). *Future teachers' interest, competence, and perception of utility value regarding computational thinking in teaching science.* Paper presented at American Educational Research Association Annual Conference.

Kale, U., & Wang Y., (2024, October). *Interested in computational thinking? Role of competence and value in preservice teachers' interest in computational thinking.* Paper presented at the Association for Educational Communications and Technology Annual Conference, Kansas City, MO.

Mayer, R. E., & Wittrock, M. C. (2006). Problem solving. In P. A. Alexander, P. H. Winne, P. A. Alexander, & P. H. Winne (Eds.), *Handbook of educational psychology* (pp. 287-303). Erlbaum.

Mishra, P., & Koehler, M. J. (2006). Technological pedagogical content knowledge: A new framework for teacher knowledge. *Teachers College Record, 108*, 1017-1054. https://doi.org/10.1111/j.1467-9620.2006.00684.x

Next Generation Science Standards. (n.d.). *Read the standards.* https://www.nextgenscience.org/search-standards

Polya, G. (1957). *How to solve it: A new aspect of mathematical method* (2nd ed.), Princeton, NJ: Princeton University Press.

Robotics Online Marketing Team (2017, December 12). Robotics in agriculture: Types and applications. *Association for Advancing Automation* [Blog]. https://www.automate.org/blogs/robotics-in-agriculture-types-and-applications

Rich, K. M., Yadav, A., & Schwarz, C. V. (2019). Computational thinking, mathematics, and science: Elementary teachers' perspectives on integration. *Journal of Technology and Teacher Education, 27*(2), 165-205. http://dx.doi.org/10.70725/303733vqleui

Tikva, C., & Tambouris, E. (2021). Mapping computational thinking through programming in K-12 education: A conceptual model based on a systematic literature Review. *Computers & Education*, *162*, Article 104083. https://doi.org/10.1016/j.compedu.2020.104083

Tissenbaum, M., Sheldon, J., & Abelson, H. (2019). From computational thinking to computational action. *Communications of the ACM, 62*(3), 34-36. https://doi.org/10.1145/3265747

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology, 25*(1), 127-147. https://doi.org/10.1007/s10956-015-9581-5

Wilson, B., DeBruin, J., Higgins, A., Gross, T., Gum, H., & Cannon,T. (2017). *Bringing our farms into focus: Specialty food crops, farm viability and market profitability in West Virginia.* Food Justice Lab. Retrieved April 15, 2025, from http://foodjustice.wvu.edu/wp-content/uploads/2018/12/WV-Farm-viability_study_final.pdf

West Virginia University Office of Data and Analytics. (2024, December 1). *Enrollment overview.* https://dataoffice.wvu.edu/reports-analytics/enrollment-overview-board

Wing, J. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A*, *366*, 3717-3725. https://doi.org/10.1098/rsta.2008.0118

Yadav, A., Stephenson, C., & Hong, H. (2017). Computational thinking for teacher education. *Communications of the ACM. 60*. 55-62. https://doi.org/10.1145/2994591

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. https://doi.org/10.1080/08993408.2018.1560550

About the Authors

Ugur Kale is an Associate Professor in Indiana University's Learning Design and Adult Education Department. His teaching focuses on instructional design, technology integration, and science education. He also develops and researches innovative approaches that prepare educators to integrate computational thinking.

Yuanhua Wang is an Assistant Professor of Science Education in West Virginia University. Her teaching focuses on elementary science education and mathematics education. She researches the implementation of inquiry-based methods in science education as well as STEM teacher knowledge and computational thinking.

Acknowledgement

Activities described in this publication were produced with the assistance of a grant from the National Science Foundation (Award #2451843; #2142274).

Sharing & Modification Permissions