2023

Project InTERSECT CT Badge Facilitator's Guide

Terrie M. Galanti
University of North Florida

Project InTERSECT

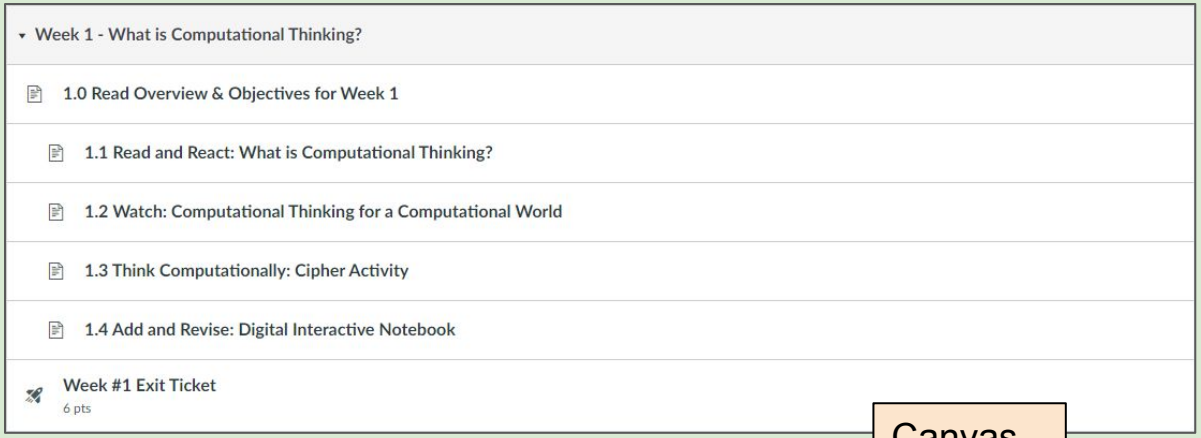
CT Facilitator Guide

Designer: Dr. Terrie Galanti
terrie.galanti@unf.edu

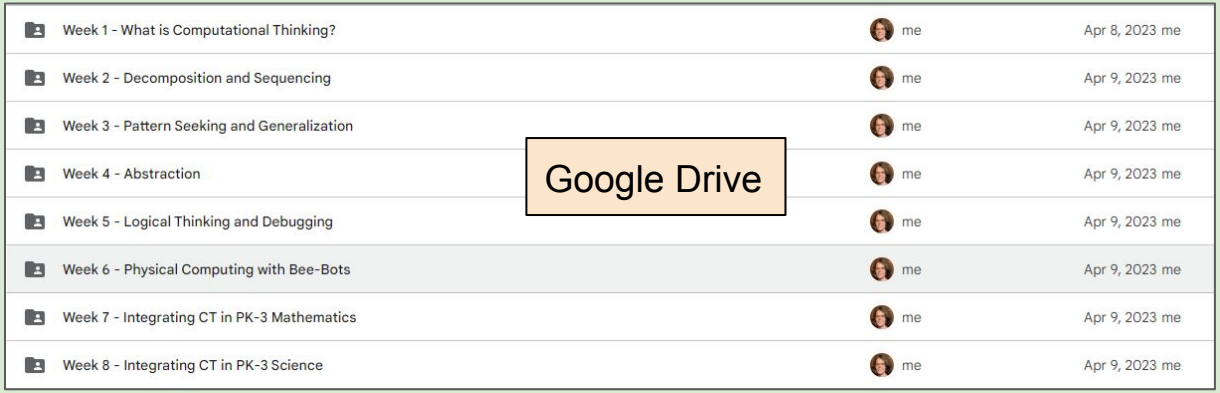
Link to CT Badge Course Description

Link to Computational Thinking Badge Materials to Copy

Badge
Canvas Course Overview



Canvas



Google Drive

1. Each page has a number for easy reference by facilitator and teacher
2. Each page title has an action verb to preview expectations.
3. Each module ends with a automatically graded “quiz”* as a self-assessment of page completion.
4. Teachers interact with the facilitator and course content through a shared Google folder.

Exit Ticket Screenshot - Instructions

Week #1 Exit Ticket

Due No due date **Points** 6 **Questions** 6 **Time Limit** None
Allowed Attempts Unlimited

Instructions

Questions #1 - #5 in this "quiz" are self-assessments of your completion of each of the tasks for the Week #1 module. Please reflect on the quality of your work as you answer each question and submit by **Monday at 11:59pm**.

If you answered "No" to any of the questions, you are welcome to continue working and to resubmit your ticket when you have completed the work.

Questions #6 is an opportunity for you to reflect on your growth as a computational thinker.

Facilitator Feedback

We will provide feedback on your work using the Google comment feature in your Digital Interactive Notebook by **Sunday at 11:59pm**. Please respond to all questions or prompts to revise your work. Our shared goal is for you to improve your understanding.

We understand that computational thinking is complicated, and we are on a collaborative journey to make sense of these new ideas.

Exit Ticket Screenshot

Example “Yes/No” question

Question 1 1 pts

I read the introduction of Williams (2020) *No Fear Coding* and provided a thoughtful response to the question in our Week #1 Shared Class Slides.

- ○ Yes
- ○ No

Exit Ticket Screenshot

Example Reflection Questions

Note that these questions will also provide evidence of teacher growth/change.

Question 7 1 pts

What is **one takeaway** from the Week #1 module that you would like to share with your facilitator?

Question 8 1 pts

What is **one question** about CT that you still have after this module?

Student Name	Week #1 Exit Ticket Out of 6	Week #2 Exit Ticket Out of 7	Week #3 Exit Ticket Out of 6
Test Student	-	-	-

What does the facilitator see in the “Grades” tab?

- Yes/No questions are graded automatically.
- Reflection questions must be scored by the facilitator.
- The facilitator can add comments to respond to teachers’ questions and ideas.
- The teacher must earn 100% on all tickets before the facilitator authorizes award of the course badge.

Weekly Facilitator Expectations

Weekly Expectations

By 8:00 am Tuesday morning

- Confirm that module is available to teachers
- Send announcement with summary of activities for the week ahead

By Tuesday evening

- Send email reminders to any student who has not submitted the quiz for the previous module

By Sunday evening

- Comment on previous week's activities in teacher DINbs using the prompts in this guide as a starting place.

Suggested Announcement Text

Welcome Message

Good morning Project InTERSECT teachers!

We are excited for you to begin this course in the Project InTERSECT badge sequence. We will be exploring what it means to think with a computational mindset. You will learn how to engage students in rich, collaborative problem solving that leverages the power of computers.

Please review the "Start Here" module. This module will explain the expectations of the course. We would also like you to familiarize yourself with with **Computational Thinking Course Materials** folder on Google Drive. You have received an invitation to access this folder. You may add it to your own Google Drive. This folder will contain all course materials including your digital interactive notebooks and weekly module slides.

Modules will be made available every Tuesday at 8:00am with an accompanying Canvas email summarizing expectations and due dates. You must complete the associated module exit ticket by the following Monday at 11:59pm. Your facilitators, will provide feedback on your exit tickets and DINbs each week.

Our first module will open **Tuesday, XXXX** at 8:00am.

Looking forward to a course filled with computational thinking and collaborative work! If you have any questions, please contact XXXX

Welcome to Week 1!

If you are a fan of the movie series “National Treasure,” this week will be fun for you! While you may not discover fabulous wealth after decrypting encoded messages, you will develop a fabulous wealth of skill and knowledge as you start to think computationally.

Be sure to complete each page in the Week 1 Canvas module in sequence starting with "1.0 Read Overview and Objectives for Week 1". Plan to spend at least 90 minutes on the Week 1 Activities. Note that your first discussion post (in Google Slides) for Week 1 is due on **Friday, XXXX** at 11:59pm.

Submit your exit ticket by **Monday, XXXX** at 11:59pm.

Feel free ask questions early and often! We are here to support your success.

Welcome to Week 2!

In this module, you will have the opportunity to discuss how computational thinking and coding are alike and how they differ. You will then get to create some digital art by writing an algorithm. The acronym STEM is often paired with Art to generate another acronym "STEAM". You will go full STEAM ahead as you channel Picasso with your “robot” apprentice and create a sequence of commands that results in your own artwork. Then you will try following a classmate’s algorithm to reveal her design.

Be sure to write your algorithm by Friday, XXXX so that your robot groupmate can execute it’s steps by **Monday, XXXX**.

Your definition of computational thinking should continue to evolve while you work on the content for this week. Please don't hesitate to reach out with questions!

Welcome to Week 3!

This week you will learn multiple ways to teach about algorithms in early childhood education. You will experience “tinkering” as a way to make sense of how a Scratch computer program works. You will also think about how computers use algorithms to search for information.

There will be MANY opportunities to build your creativity and confidence with CT this week! Enjoy, and be ready to ask lots of questions!

Welcome to Week 4!

This week you will focus on the idea of abstraction as one of six computational thinking skills we can emphasize in early childhood education. You will continue “tinkering” as a way to make sense of how a Scratch computer program works. Your Scratch project this week might look like lots of coding blocks.... you will want to focus on the top level program as you think algorithmically about what all of the blocks might be doing.

Your work for this week is due on **Monday XXXX.** Our goal is for your struggle to be productive - don't hesitate to reach out to either of us if you have computational thinking questions!

If you have not already done so, create a Scratch account so that you can share the link to your modified Scratch program. You will need to "remix" Dr. Galanti's program and share a link. The module has a video explaining this process.

Have a great week!

Welcome to Week 5!

This week you will engage in logical thinking as another of six computational thinking skills we can emphasize in early childhood education. Enjoy solving puzzles and using your debugging skills to find errors in a Scratch program.

Your work for this week is due on **Monday, XXXX.** We appreciate the time and energy you are investing in these modules. As schools continue to think about what computer science can look like for our youngest learners, you are developing specialized knowledge that you can share!

Welcome to Week 6!

Good afternoon computational thinkers!

Now that you have experienced each of our computational thinking skills (decomposition, algorithmic thinking, abstraction, generalizing and patterns, and logic), you will begin thinking about how to design integrated CT experiences for your future students that connect to the subjects they are learning (e.g., ELA, mathematics and science).

This week you will design a classroom activity for Bee-Bot, a wonderful early childhood CT resource. Note that you can check out Bee-Bots from our UNF STEP lab to use for your field or internship experiences. You will also explore a wonderful K-2 computer science curriculum to inspire you in your CT instructional design. You will see lessons with Bee-Bots, ScratchJr, and unplugged CS.

Your work for this week is due on **Monday, XXXX.** Happy programming and designing!

Welcome to Week 7!

As you count down the days until the end of our CT badge course, we hope you enjoy new ways to count with the adding 1-100 activity. You will find that CT is not just for "squares" as you define and draw shapes in Scratch.

You will also think deeply what it means to integrate CT in mathematics lessons. There are many conceptual overlaps between CT and mathematics. You may be able to find the ideas of decomposition and algorithmic thinking in a lesson in your school's curriculum, or you find to use patterns and generalization to enhance the mathematical reasoning in a lesson. Finally, you may be able to use Bee-Bots or Scratch to extend the mathematical thinking opportunities with computers!

Please complete all tasks by **Monday, XXXX**.

Welcome to Week 8!

Good morning computational thinkers!

The finish line is in sight! Your race to understanding CT concepts has been full of challenges. In many cases you exceeded our expectations (we aim high!) with your insight and thoughtful responses. This week, you will switch from a focus on mathematics to a focus on science.

Your magnetic and electric personalities will light up the CT fires in a coding activity, and you will see opportunities to see CT within a classic "Oh Dear" science lesson. You will also learn how spreadsheets can both enhance and extend this science lesson. Finally, you will reflect on your experiences with this badge by reflecting on the slides in your DINb and by sharing your perspectives on learning CT in a Flip video.

Please complete all tasks by **Monday, XXXX.**

Thanks for your all of your efforts in this badge course - we are excited about how you will share your new CT knowledge with our youngest learners.

Weekly Facilitator Checklist

- ❑ E-mail students who have not submitted the previous week's quiz
- ❑ Add comments to individual slides using suggestions in this guide
- ❑ Respond to teacher comments and questions on Canvas exit ticket.

Digital Interactive Notebook (DINb)

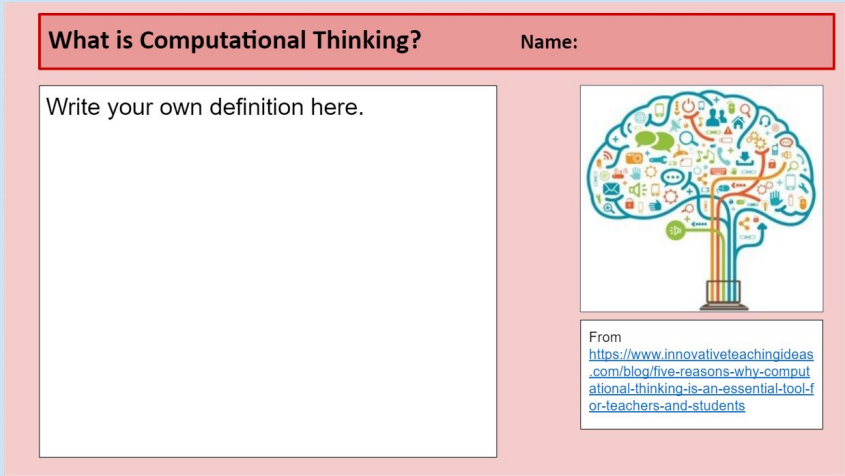
Week #1
What is computational thinking?

Week #1 Facilitator Guide - Complete by Sunday of Week #2

Add the following **comments** to the "What is Computational Thinking?" individual slide in the DINb

Begin to think about how your definition captures the relationship between computational thinking and computers. We are solving problems, but in a specific way! We will continue to think about this definition.

Remember there are no right or wrong answers as we are developing our CT understandings. You can revise this definition as you have more experiences with computational thinking!



Week #1 Facilitator Guide - Complete by Sunday of Week #2

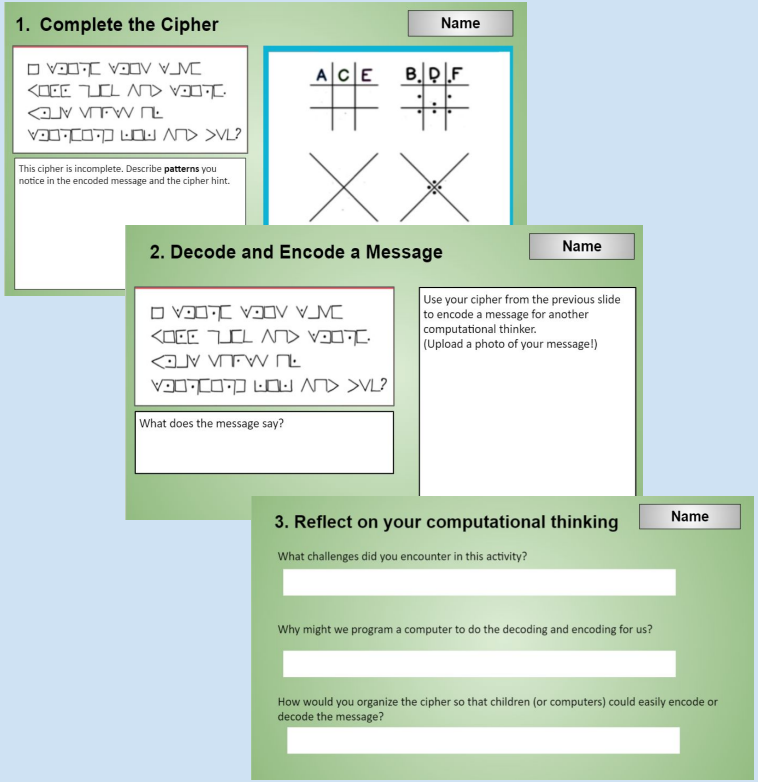
Slide 1 Comment Ideas

You used the CT skills of pattern seeking and problem solving! How did you decide where to place the S, T, U, and V?

What connections do you see between language arts and your thinking about encoding and decoding in this activity?

Slide 2 Comment Ideas

We can think of computer programming as "coding" by humans,
and then the computer "decodes" our ideas.
I am still working on decoding your message! We love computers because they can do the work much more efficiently than humans can.



Week #1 Facilitator Guide - Complete by Friday of Week #2

Other Comment Ideas:

- *What did you notice about this message before you began to complete the cipher? This initial computational thinking helps us to decompose challenging problems.*
- *Pattern seeking is a vital computational skill - I am imagining how we could ask a computer to "narrow" the possibilities for us to build a cipher based on patterns.*
- *What connections do you see between language arts and your thinking about encoding and decoding in this activity?*
- *Notice that debugging doesn't simply occur at the end of an activity; you used it to confirm the correctness of your cipher!*

Week #1 Facilitator Guide - Complete by Friday of Week #2

Other Comment Ideas:

- *Your definition captures the important idea that rich CT can happen with and without computers!*
- *Keep in mind that computers don't solve problems; people build and instruct computers so that they can the solve problems!*
- *You captured the important idea that we have to do the reasoning and the problem solving that allow us to make use of the power of computers.*
- *How would you leverage patterns to get computers to solve a problem for you?*

Week #2
Decomposition and Algorithmic Thinking

Week #2 Facilitator Guide - Complete by Friday of Week #3

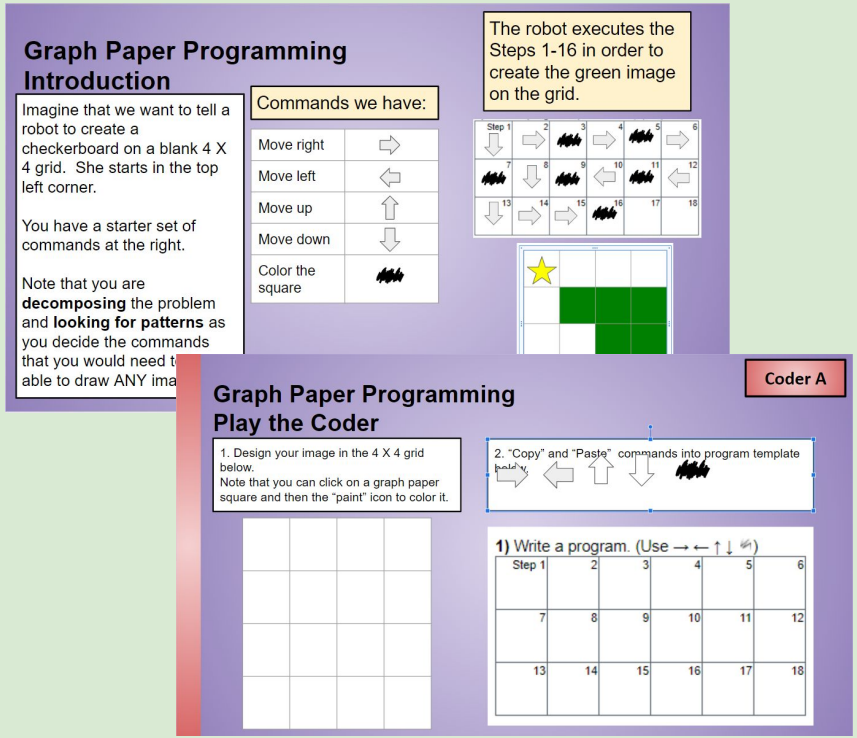
Play the Coder Slide **Comment Ideas**

Look for sections in your code where you see repeating patterns of the same commands or sets of command repeating. When we code we can use "loops" to shorten number of commands.

If I run your same program twice in a row, I will get two different images. Why might that be?

I notice that your code has your robot move in a clockwise motion. We could challenge our young learners to write a second program that would ask the robot to move counterclockwise :) We will teach them that CT is often about multiple solutions to a problem.

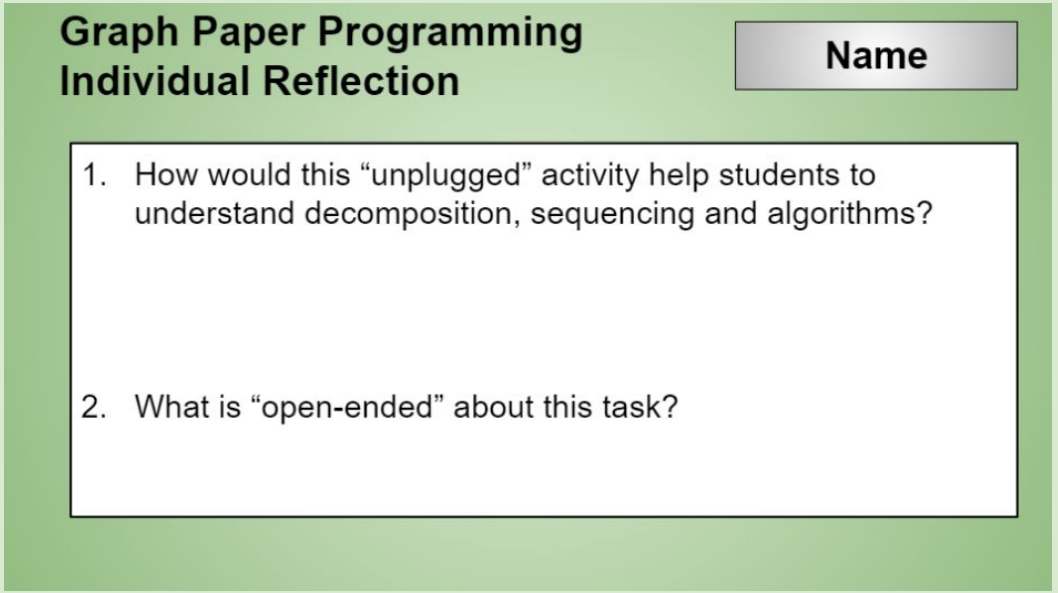
You are thinking like a human who knows that the robot can only interpret 5 commands. We can't say things like "color the top and bottom rows".



Week #2 Facilitator Guide - Complete by Sunday of Week #3

Comment Idea

"Notice that the activity is open-ended because there is more than one way to sequence the commands to draw an image!"



Graph Paper Programming
Individual Reflection

Name

1. How would this "unplugged" activity help students to understand decomposition, sequencing and algorithms?

2. What is "open-ended" about this task?

Week #3
Pattern Seeking and Generalization

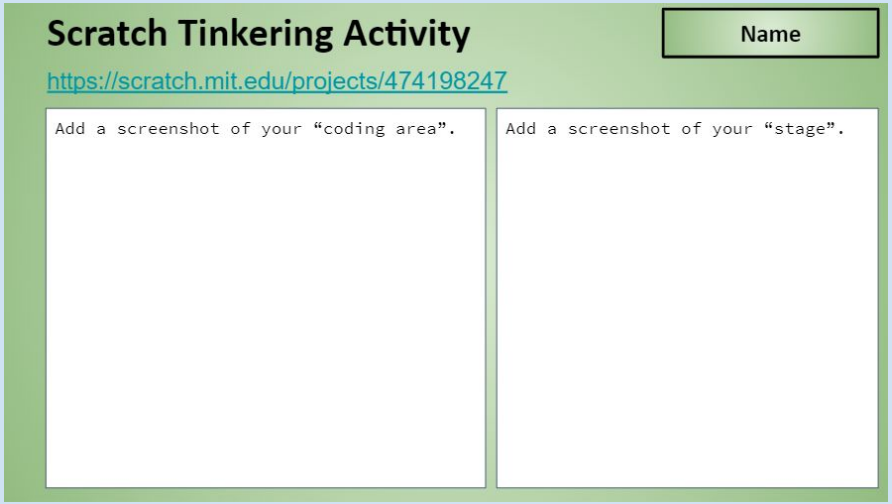
Week #3 Facilitator Guide - Complete by Sunday of Week #4

Make a positive **comment** about what the student created in Scratch!

Other Comment Ideas

You are thinking computationally as you synchronize two programs that run in parallel. Explore the yellow event blocks in Scratch to see how else you can "trigger" the start of a program.

Notice that wait time is how we synchronize the programs associated with each of our sprites.



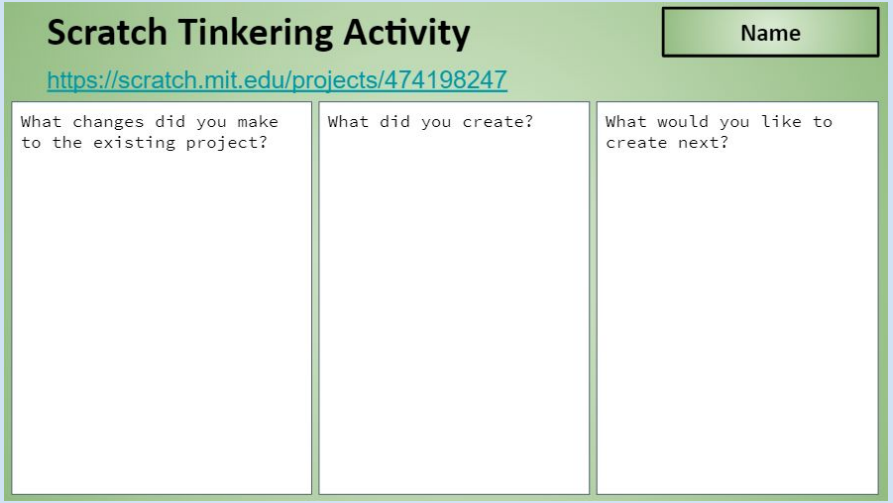
Week #3 Facilitator Guide - Complete by Sunday of Week #4

Comment Ideas:

How does a tinkering approach to coding make computational thinking more accessible to your students?"

Tinkering will be a foundational idea in all of our STEM work together - the deepest learning can happen when we are not trying to follow directions or read manuals!

You are seeing the power of coding as expression - we do not always have to use a computer to get the answer to a problem.



Scratch Tinkering Activity

Name

https://scratch.mit.edu/projects/474198247

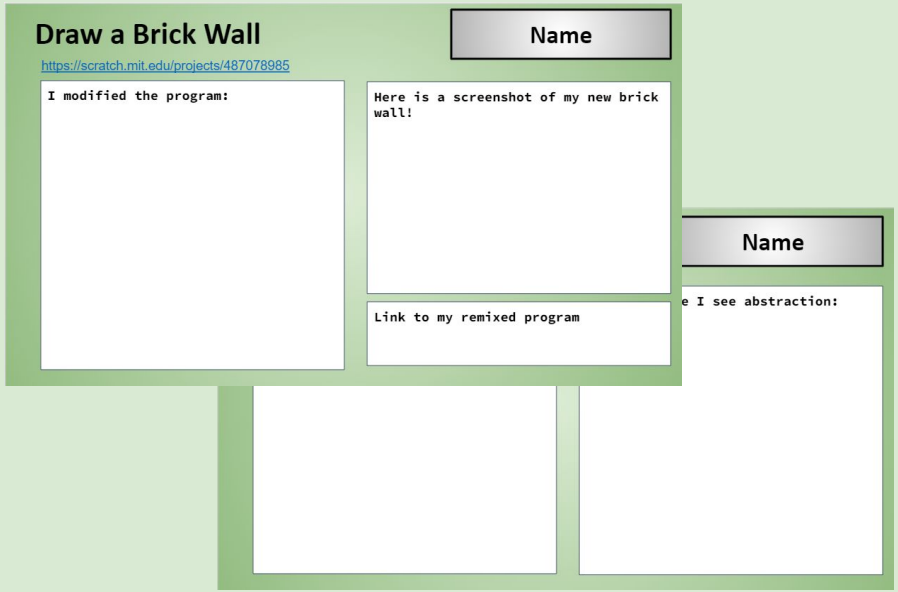
What changes did you make to the existing project?	What did you create?	What would you like to create next?

Week #4
Abstraction

Week #4 Facilitator Guide - Complete by Sunday of Week #5

Comment on an interesting addition or another ideas for adding functionality to the program.

Note: Be sure to click on the program link and execute by clicking on the green flag! Tell teachers what you saw…

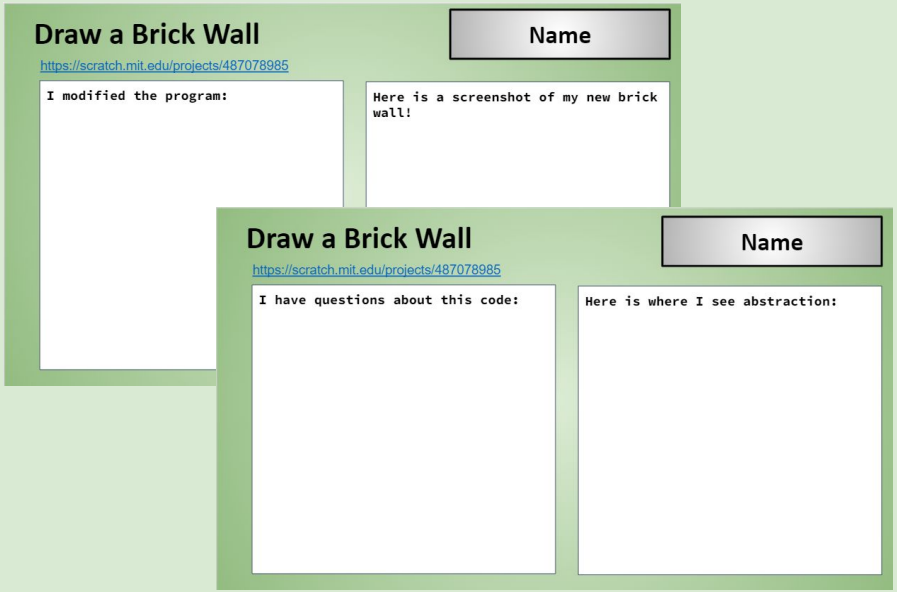


Week #4 Facilitator Guide - Complete by Sunday of Week #5

Comment Ideas

There was one block that built the wall and you could decide the number of rows. All of the other functions were "hidden" in blocks underneath your code. This is an example of abstraction.

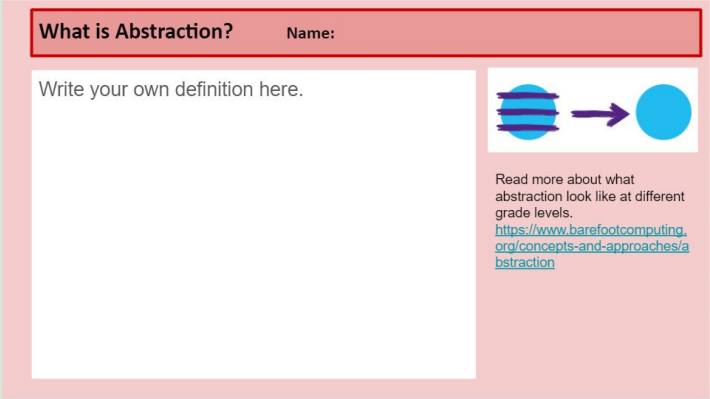
There are many layers of abstraction in computer programs! Imagine how many blocks it would require to build this wall if we did not use abstraction!!!



Week #4 Facilitator Guide - Complete by Sunday of Week #5

Add the following **comment** to the “What is abstraction?” slide.

We can think of every computer programming language and every app on our phone as an abstraction - we are hiding the complexity of the computer hardware and electrical signals underneath. We will also talk later in the course about simulations as abstractions.



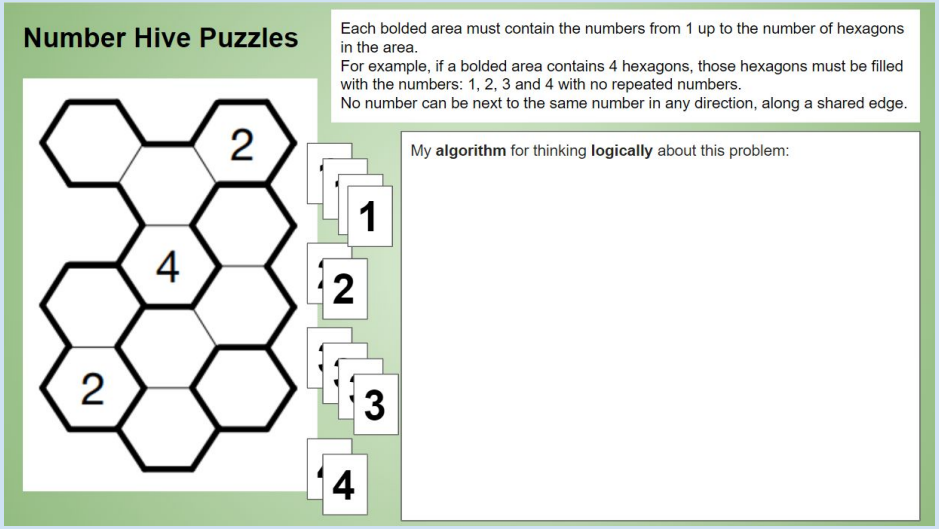
Week #5
Logical Thinking and Debugging

Week #5 Facilitator Guide - Complete by Sunday of Week #6

Comment Ideas

How would you respond to the comment: “Logical puzzles aren’t creative because there is only one right answer.”?

We can use our algorithm to generalize and program a computer to solve a puzzle with many more hexagons!



Week #6
Physical Computing with Bee-Bots

Week #6 Facilitator Guide - Complete by Sunday of Week #7

Comment:

Add the following question to the leftmost block. “Suppose we added a “spin” command to the Bee-Bot that result a 360 degree turn. How would that command represent both algorithmic thinking and abstraction?”

Note: If the student does not include words like “go home”, “turn left”, “move forward”, provide guidance.

Week #6 Facilitator Guide - Complete by Sunday of Week #7

Comment Ideas

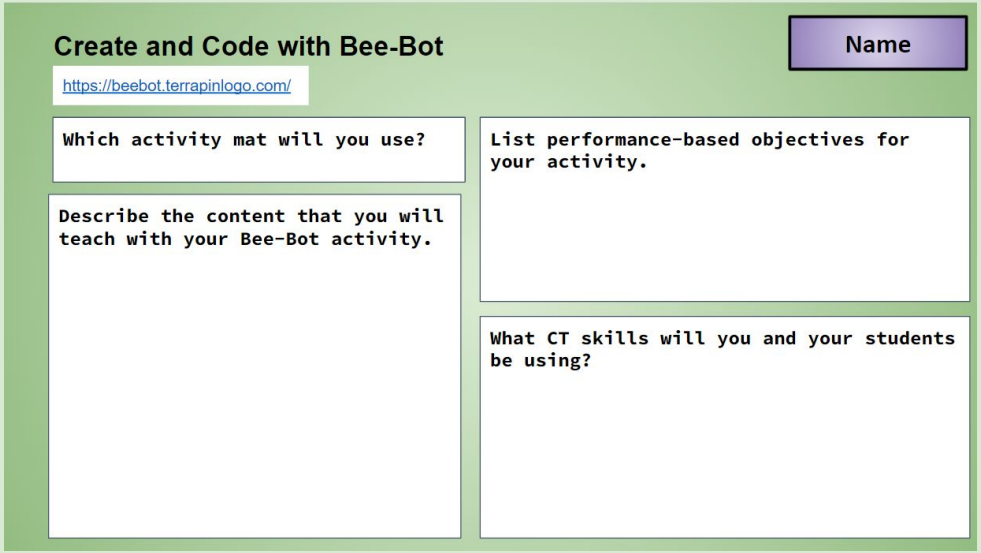
When I execute your algorithm I spell "pig". Was that your goal?

You have written a program with 25 commands to spell a word with 4 letters. Imagine a student says, "Wow, your program is better than mine because I only spelled "cat". How would you respond?

Week #6 Facilitator Guide - Complete by Sunday of Week #7

Add the following **comment**:

Notice that you are integrating computer science and (fill in content area). How might CT help students to understand this content more deeply?



Week #7
Integrating CT in PK-3 Mathematics

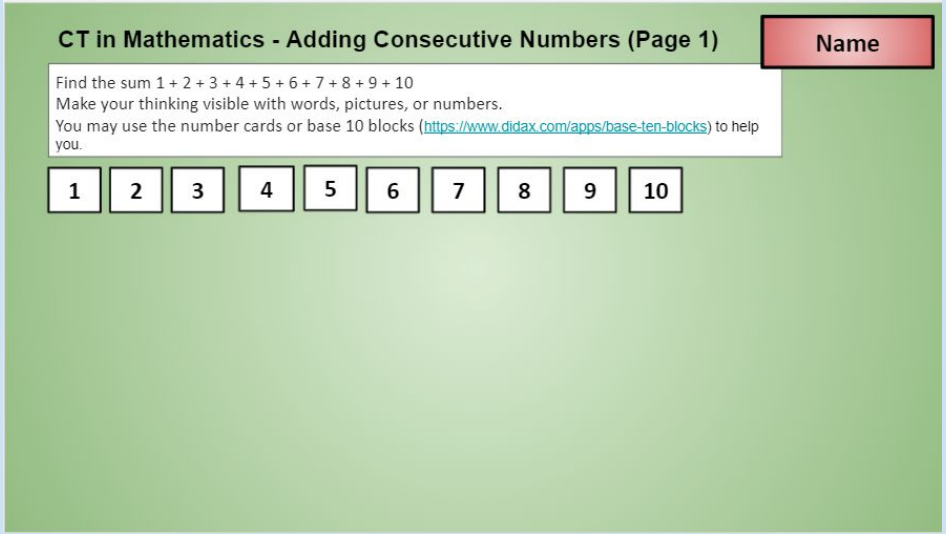
Week #7 Facilitator Guide - Complete by Sunday of Week #8

Comment Ideas

*Notice that offering choices of physical manipulatives **enhances** the lesson by making computational thinking visible.*

Recognizing patterns prepares us to be able to use conditionals and loops in our programming.

Students are also learning to persist through iteration and experimentation and accept that failure and struggle are a part of the learning process.
https://equip.learning.com/pattern-recognition-computational-thinking/



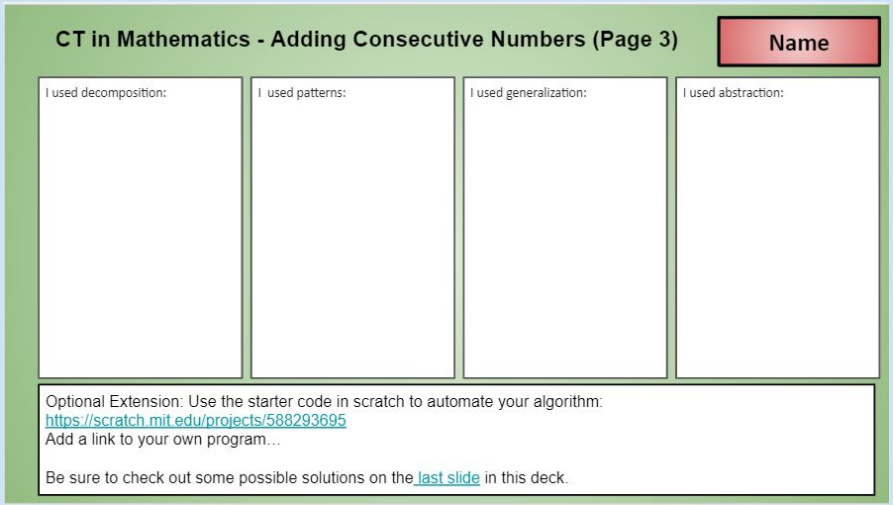
Week #7 Facilitator Guide - Complete by Sunday of Week #8

Comment Idea

The opportunity to write a Scratch program **extends** the lesson. Students can see how their mathematical thinking can translate into CT.

What questions do you have about Dr. Galanti's two Scratch programs to find the sum from 1 to 10?
https://scratch.mit.edu/projects/500974174

https://scratch.mit.edu/projects/500975232



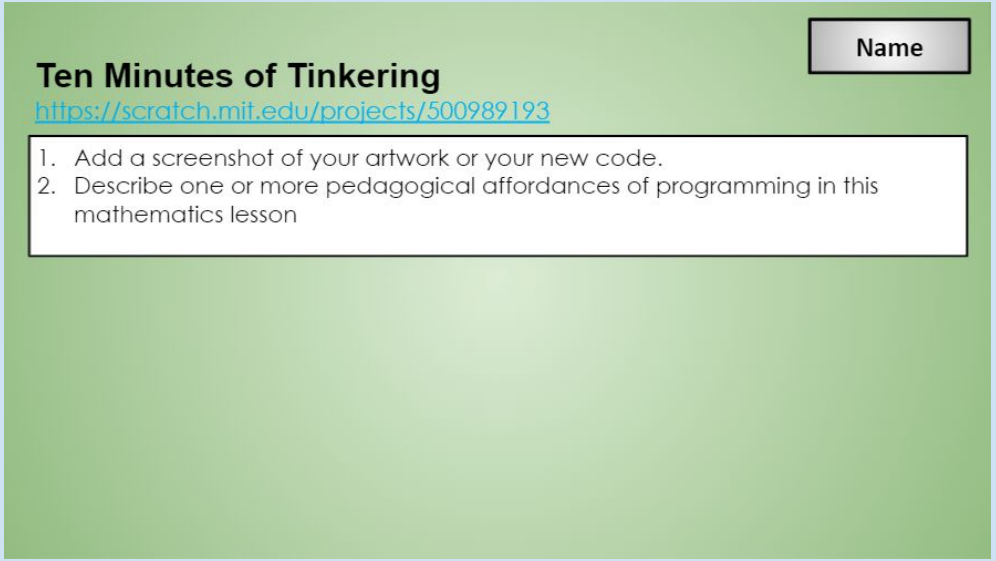
Week #7 Facilitator Guide - Complete by Sunday of Week #8

Comment Idea

How does this Scratch program deepen students understanding about squares?

Students can connect the definition of a square to their code. The "draw square" block is an accessible example of abstraction - students will need to understand which parameters they can change and which they cannot.

How does this activity build a child's sense of wonder and surprise?



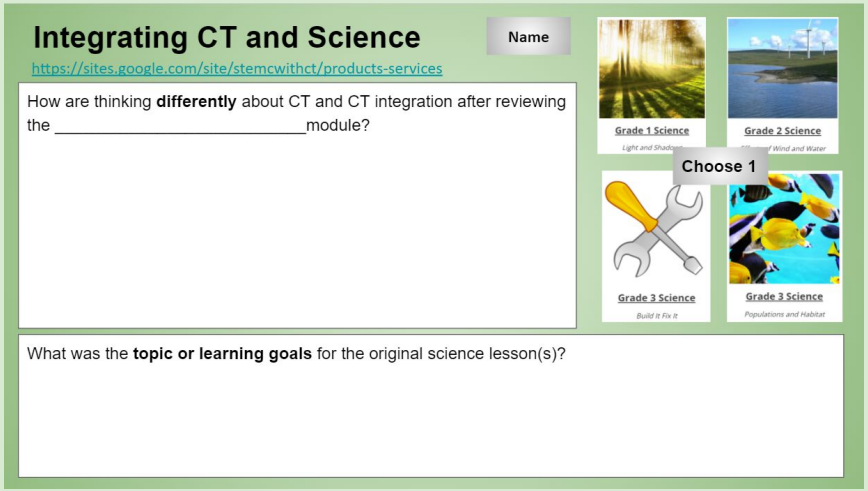
Week #8
Integrating CT in PK-3 Science

Week #8 Facilitator Guide - Complete by Sunday of Week 9

Comment Ideas

The computer creates a graph based on data. Questions about the graph increase opportunities for pattern seeking and generalization.

There is abstraction in the "physical" model with children because it hides the complexity of a habitat...the model on a computer is an enhanced CT opportunity.



Week #8 Facilitator Guide - Complete by Sunday of Week #8

Comment Ideas

There is abstraction in the "physical" model with children because it hides the complexity of a habitat...the model on a computer is an enhanced CT opportunity.

What challenges do you anticipate you would face as a classroom teachers who wanted to implement this sequence of lessons?