CT IN ELEMENTARY SCIENCE TEACHER EDUCATION

Developing a Framework for Integration

Dr. Diane Jass Ketelhut & Lautaro Cabrera
University of Maryland, College Park

Our Project

- CT Integrated into Elementary Science Methods Course
  - *Focused on four sessions*
  - *Final assignment: CT-infused science lesson*
  - *Modest results: preservice teachers used CT terms loosely*
- Professional Development experience
  - *Science Teaching Inquiry Group in Computational Thinking **(STIG^{CT})***
  - *Pre-service and in-service teachers learn and work together, including mentor-mentees pairs*
  - *Researchers and teachers co-design CT-infused science lesson plans*
  - *Introduce teachers to CT concepts through elementary school science activities*

Framework Development

- For both the course and the STIG^{CT}, we iteratively developed a framework for integrating CT into elementary science.
- The framework guided participant learning, discussion around CT, and integration of CT into lesson plans.
- Different versions of the framework were accompanied by different results in how teachers integrated CT

FRAMEWORK YEAR 1

Our Framework Iterations: Year 1

- Drew from multiple sources:
  - *Weintrop et al. (2016): CT practices specifically for science and math*
  - *CSTA & ISTE (2011): inclusion of dispositions and attitudes*
  - *Barr & Stephenson (2011): use of concrete examples*
- Created our own examples of each CT Practice (from Weintrop et al.)

Weintrop et al. (2016)
Data Practices
Collecting Data
Creating Data
Manipulating Data
Analyzing Data
Visualizing Data
Modeling & Simulation Practices
Using Computational Models to Understand a Concept
Using Computational Models to Find and Test Solutions
Assessing Computational Models
Designing Computational Models
Constructing Computational Models
Computational Problem-Solving Practices
Preparing Problems for Computational Solutions
Programming
Choosing Effective Computational Tools
Assessing Different Approaches/Solutions to a Problem
Developing Modular Computational Solutions
Creating Computational Abstractions
Troubleshooting and Debugging
Systems Thinking Practices
Investigating a Complex System as a Whole
Understanding the Relationships within a System
Thinking in Levels

CSTA & ISTE (2011)

B Operational Definition

ISTE and CSTA collaborated with leaders from higher education, industry, and K–12 education to develop an operational definition of CT. The operational definition provides a framework and vocabulary for CT that will resonate with all K–12 educators.

CT is a problem-solving process that includes (but is not limited to) the following characteristics:

- Formulating problems in a way that enables us to use a computer and other tools to help solve them
- Logically organizing and analyzing data
- Representing data through abstractions such as models and simulations
- Automating solutions through algorithmic thinking (a series of ordered steps)
- Identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources

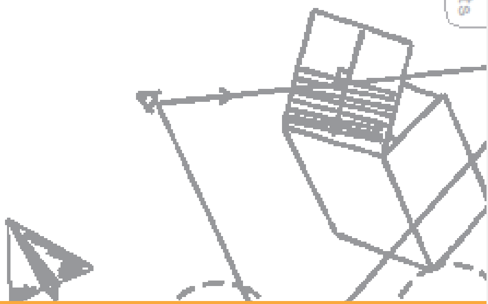


Table of Contents

C CT Vocabulary and Progression Chart

	Definition	Grades PK to 2	Grades 3 to 5	Grades 6 to 8	Grades 9 to 12
Data Collection	The process of gathering appropriate information	Conduct an experiment to find the fastest toy car down an incline and record the order of cars across the finish line in a chart.	Review examples of writing to identify strategies for writing an essay.	Design survey questions to gather appropriate information to answer questions (e.g., asking fellow students if they were absent from school in the past month and whether they were suffering from the flu).	Students develop a survey and collect both qualitative and quantitative data to answer the question: "Has global warming changed the quality of life?"
Data Analysis	Making sense of data, finding patterns, and drawing conclusions	Make generalizations about the order of finishing a toy car race based on the characteristics of the car with a focus on weight. Test conclusions by adding weight to cars to change results.	Categorize strong and weak examples of writing samples to develop a rubric.	Produce and evaluate charts from data generated by a digital probe and describe trends, patterns, variations, and/or outliers represented in the chart.	Use appropriate statistical methods that will best test the hypothesis: "Global warming has not changed the quality of life."
Data Representation	Depicting and organizing data in appropriate graphs, charts, words, or images	Create a chart or a line drawing that shows how the speed of a toy car changes when its weight is changed.	Match each writing sample to the rubric and create a chart showing which example best fits in each category of the rubric.	Plot data using different charting formats and select the most effective visual representation strategy.	Groups of students represent the same data in different ways based on a position relating to the question: "Has global warming changed the quality of life?" Different representations may result in varying conclusions.
Problem	Breaking down	Create directions to a location	Develop a plan to make the	In planning the publication of a	Consider the large-scale problem: "What

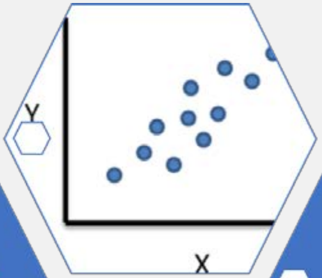
Table of Contents

Year 1 “Framework” Challenges

- The framework language was sometimes inaccessible or overwhelming for teachers—it was based on CS terminology
  - *E.g., <u>algorithmic thinking</u> or <u>computational abstraction</u>*
- Hard to differentiate CT practices from other more common scientific practices
  - *E.g., CT data collection vs. science data collection*

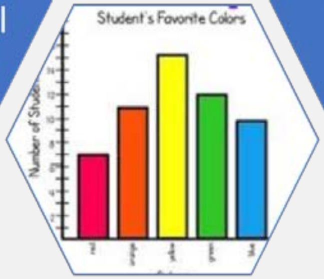
FRAMEWORK YEAR 2

Using Data



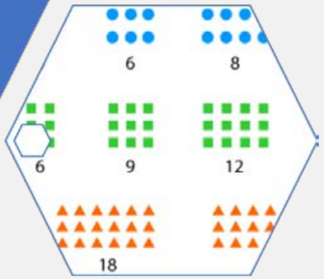
Collecting data with computational devices

Finding patterns and relationships in data sets

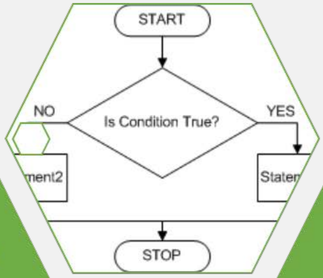


Sorting data

Creating graphs or charts



Programming



Breaking down problems into smaller parts

Creating step by step instructions to solve a problem

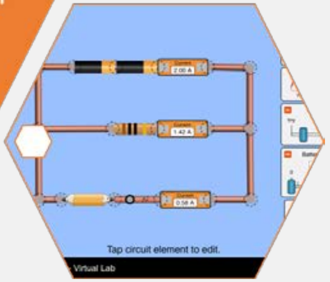
Coding

Test, Adjust to Improve, Retest, Readjust to Improve…

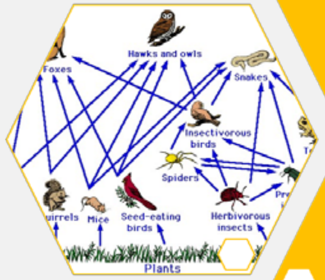
Computational Simulations

Using computational simulations

Assessing computational simulations



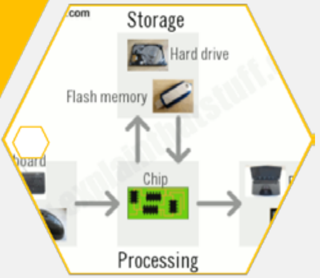
Systems Thinking from a CT Perspective



Identifying quantifiable parts of a system

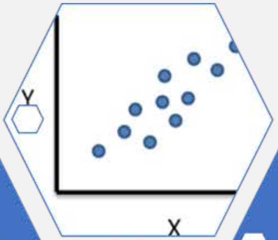
Considering numerical relationships within a system

Considering how changes to the quantifiable parts contribute to results of the system

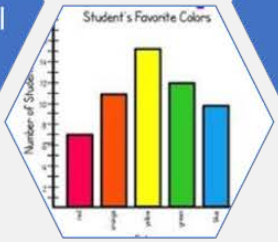


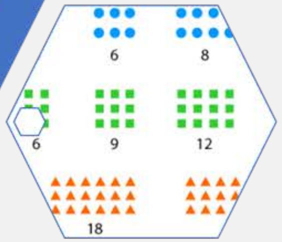
Unified sources into one framework

Using Data

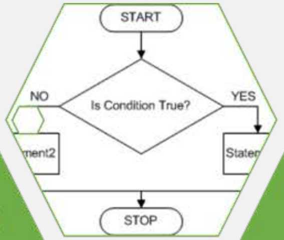


- Finding patterns and relationships in data sets
- Creating graphs or charts
- Collecting data with computational devices
- Sorting data





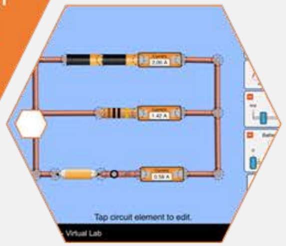
Programming



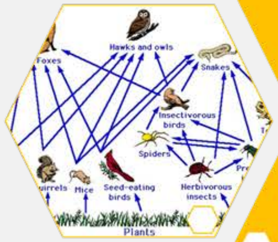
- Breaking down problems into smaller parts
- Test, Adjust to Improve, Retest, Readjust to Improve…
- Coding
- Creating step by step instructions to solve a problem

Computational Simulations

- Assessing computational simulations
- Using computational simulations



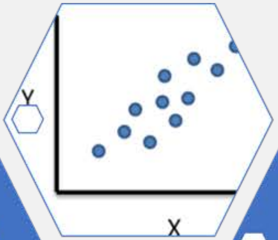
Systems Thinking from a CT Perspective



- Identifying quantifiable parts of a system
- Considering how changes to the quantifiable parts contribute to results of the system
- Considering numerical relationships within a system

Reduced number of practices

Using Data

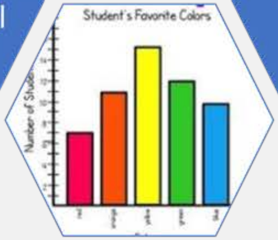


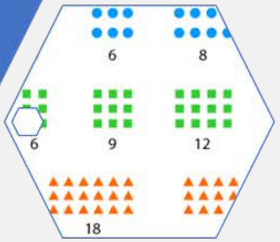
Finding patterns and relationships in data sets

Creating graphs or charts

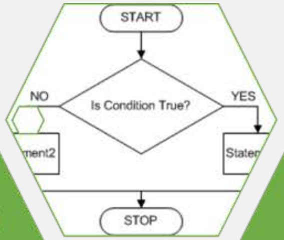
Collecting data with computational devices

Sorting data





Programming



Breaking down problems into smaller parts

Test, Adjust to Improve, Retest, Readjust to Improve...

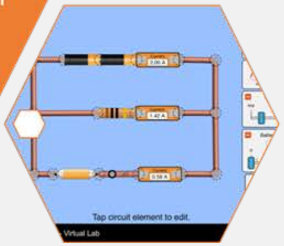
Coding

Creating step by step instructions to solve a problem

Computational Simulations

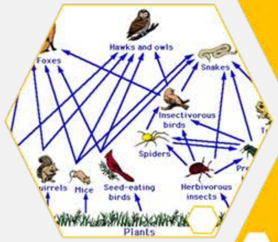
Assessing computational simulations

Using computational simulations



Systems Thinking from a CT Perspective

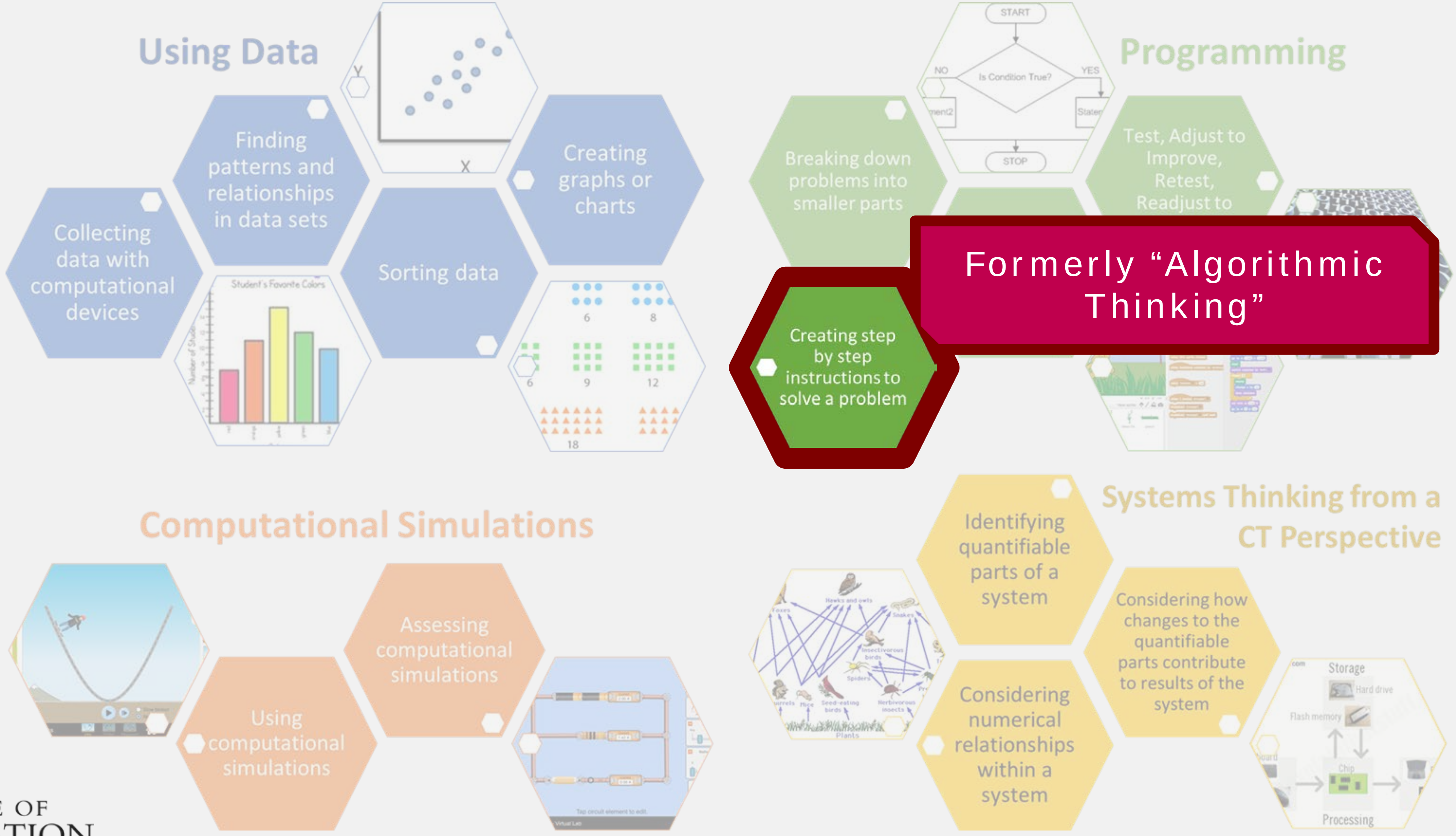
Identifying quantifiable parts of a system



Considering how changes to the quantifiable parts contribute to results of the system

Considering numerical relationships within a system

Simplified language to avoid CS jargon



Differentiated CT from science practices

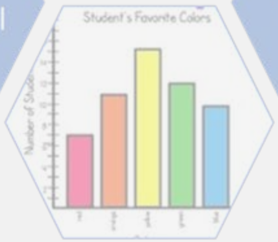
Using Data

Finding patterns and relationships in data sets

Creating graphs or charts

Collecting data with computational devices

Sorting data

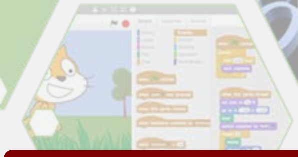


Programming

Breaking down problems into smaller parts

Test, Adjust to Improve, Retest, Readjust to Improve…

Coding

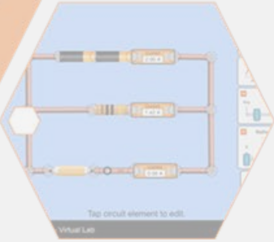


Creating step by step instructions to solve a problem

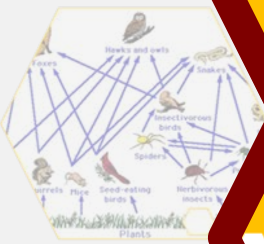
Computational Simulations

Assessing computational simulations

Using computational simulations



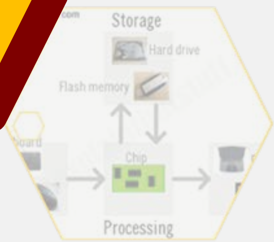
CT Perspective



Identifying quantifiable parts of a system

Considering how changes to the quantifiable parts contribute to results of the system

Considering numerical relationships within a system



Added a quantifiable or numerical component

COLLEGE OF EDUCATION
UNIVERSITY OF MARYLAND

Preliminary Results

- With the new framework, teachers are feeling more comfortable integrating CT
  - *Both in written reflections and self-efficacy measures*
- They are more successfully integrating CT into their lesson plans than in Year 1
  - *The instances of CT in their lesson plans more closely resembled the CT practices of the framework*
- Mentors and mentees are benefitting from working together
  - *Different but complementary expertise*

Remaining Challenges

- Almost no teachers integrated Systems Thinking from a CT Perspective. Are these practices appropriate for the elementary level?
- Simplifying language to avoid CS jargon may have led to some superficial uptake
  - *Sometimes "step-by-step instructions" meant following any type of procedure was considered CT*

Moving Forward

- How are teachers *implementing* the lessons they design?
  - *What are the instances of CT that are developmentally appropriate, work within school structures, and teachers feel comfortable integrating?*
- Which CT practices are making it into the Elementary classroom?
  - *How is the framework guiding the design and implementation of lessons?*

QUESTIONS?
THANK YOU!

Dr. Diane Jass Ketelhut & Lautaro Cabrera

Randy McGinnis, Jan Plane, Kelly Mills, Merijke Coenraad and Heather Killen

University of Maryland, College Park

djk@umd.edu | cabrera1@umd.edu

This material is based upon work supported by the National Science Foundation under Grant No. 1639891. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.