Computational Thinking in STEM Disciplines: Perspectives from a Mathematics Education Researcher

Elise Lockwood
Oregon State University

University of Oslo

What is Computational Thinking?

Background and Motivation

- Computing is an integral component of scientific and mathematical work
- In the US, the Next Generation Science Standards explicitly include “Using Mathematics and Computational Thinking”; “Computer Science for All”
- There seems to be growing attention on computation
  - My department has a “computational” requirement in our major
  - There are increasing job postings for “computational mathematicians”
  - There are newly-developed computational departments
  - Your university is embracing this emphasis on computation

Background and Motivation

- “By 2020, one of every two jobs in the STEM fields will be in computing” (Association for Computing Machinery, 2013)
- Some scientists believe that computing complements theory and experimentation as pillars of science (Wing, 2017)
- There is increasing evidence that there is a relationship between learning computation and learning valuable mathematical and scientific concepts and practices (Eisenberg, 2002; National Research Council, 2011; Wilensky et al., 2014)

Computational Thinking (CT)

- Goals for this talk:
- Present some existing literature and background related to CT
- Offer existing characterizations of CT, then present and unpack the working definition that I personally use
- Discuss examples of current and future research endeavors in CT

Historical Context of Computational Thinking

- In 1962, Perlis said that all undergraduates should learn to program as part of a liberal education, and Papert (1972, 1980) suggested extending programming into childhood
- In 1996, Papert introduced the term *computational thinking* to refer to “the affordances of computational representations for expressing powerful ideas” (Weintrop, et al. 2016, p. 129-130)
- In 2006, Jeannette Wing renewed the modern conversation about CT

Historical Context of Computational Thinking

- Wing initially described CT as “taking an approach to solving problems, designing systems and understanding human behaviour that draws on concepts fundamental to computer science” (2006, p. 33).
- Wing characterized CT broadly, encompassing activities such as:
  - “**thinking recursively**”
  - “**using abstraction** and decomposition when attacking a large complex task or designing a large complex system”
  - “**using heuristic reasoning** to discover a solution”
  - “**making trade-offs** between time and space and between processing power and storage capacity”

Historical Context of Computational Thinking

- Wing’s initial descriptions were quite vague and far reaching
- Grover and Pea (2013) reviewed CT literature, and they found “definitional confusion” and widely varying perspectives of CT
- “Clearly, much remains to be done to help develop a more lucid theoretical and practical understanding of computational competencies in children...It is time to redress the gaps and broaden the 21st-century academic discourse on computational thinking” (p. 42)

Perspectives on Computational Thinking

- CT as broad problem solving and critical thinking skills applied to computer science concepts (Snow, 2016)
- CT as “thinking like a computer scientist” (Wing, 2014)
- CT as “The thought process involved in formulating problems so their solutions can be represented as computational steps and algorithms” (Aho, 2012, p. 832)
- CT as “The thought processes involved in expressing solutions as computational steps or algorithms that can be carried out by a computer” (Cuny, Snyder, & Wing, 2010)

My Working Definition of Computational Thinking (Wing, 2014)

- **CT is the way of thinking that one uses to formulate a problem and/or express its solution(s) in such a way that a computer could effectively carry it out.**

My Working Definition of Computational Thinking (Wing, 2014)

- **CT is the <u>way of thinking</u> that one uses to formulate a problem and/or express its solution(s) in such a way that a computer could effectively carry it out.**
- CT is about a person's thinking: "CT is about the thought processes that one goes through before you even write some code, before you're even creating an artifact" (Wing, 2016)
- "Way of thinking" is a technical term in math education literature
- "A "way of thinking" [is] when a person has developed a pattern for utilizing specific meanings … in reasoning about particular ideas" (Thompson, et al., 2014, p. 14)

My Working Definition of Computational Thinking (Wing, 2014)

- **CT is the way of thinking that one uses to <u>formulate a problem and/or express its solution(s)</u> in such a way that a computer could effectively carry it out.**
- Computational thinking can be leveraged in problem formulation (even if a solution is not found), and it can be leveraged in expressing a solution (even if the formulation was not computational)

My Working Definition of Computational Thinking (Wing, 2014)

- **CT is the way of thinking that one uses to formulate a problem and/or express its solution(s) in such a way that a computer could <u>effectively</u> carry it out.**
- There may be multiple pathways to formulate the problem or solutions
- An ultimate goal is correctness, and efficiency must be considered

My Working Definition of Computational Thinking (Wing, 2014)

- **CT is the way of thinking that one uses to formulate a problem and/or express its solution(s) in such a way that a computer could effectively carry it out.**

Current and Future Research Endeavors in CT

- Here are three kinds of research related to CT that is currently being (and should continue to be) conducted

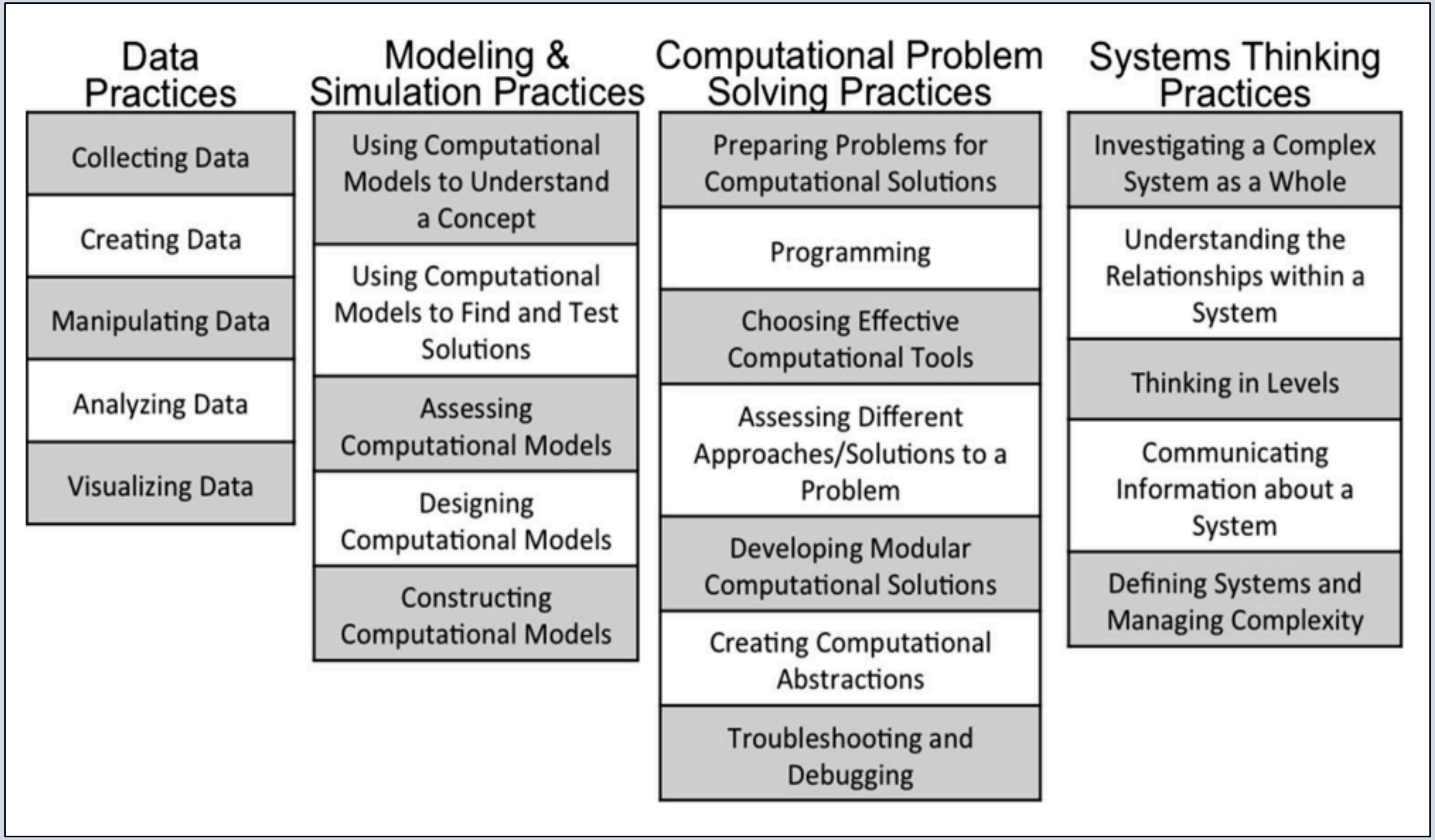
1. What is CT? – Theoretical investigations into the nature of CT itself
2. How does CT help students learn mathematical or scientific content?
3. How does CT help students develop mathematical or scientific practices?

- Be watching for a question that most interests you

1. Theoretical investigations into the nature of CT itself

- Weintrop, et al. (2016) created a taxonomy to characterize classroom practices that reflect CT
  - They looked at CT literature to identify important computational practices
  - They found lessons and activities that emphasized these skills in high school mathematics and science classrooms
  - They developed the taxonomy and validated it through interviews with experts
- They focus on high school, but similar kinds of activity may apply in undergraduate classrooms

1. Theoretical investigations into the nature of CT itself

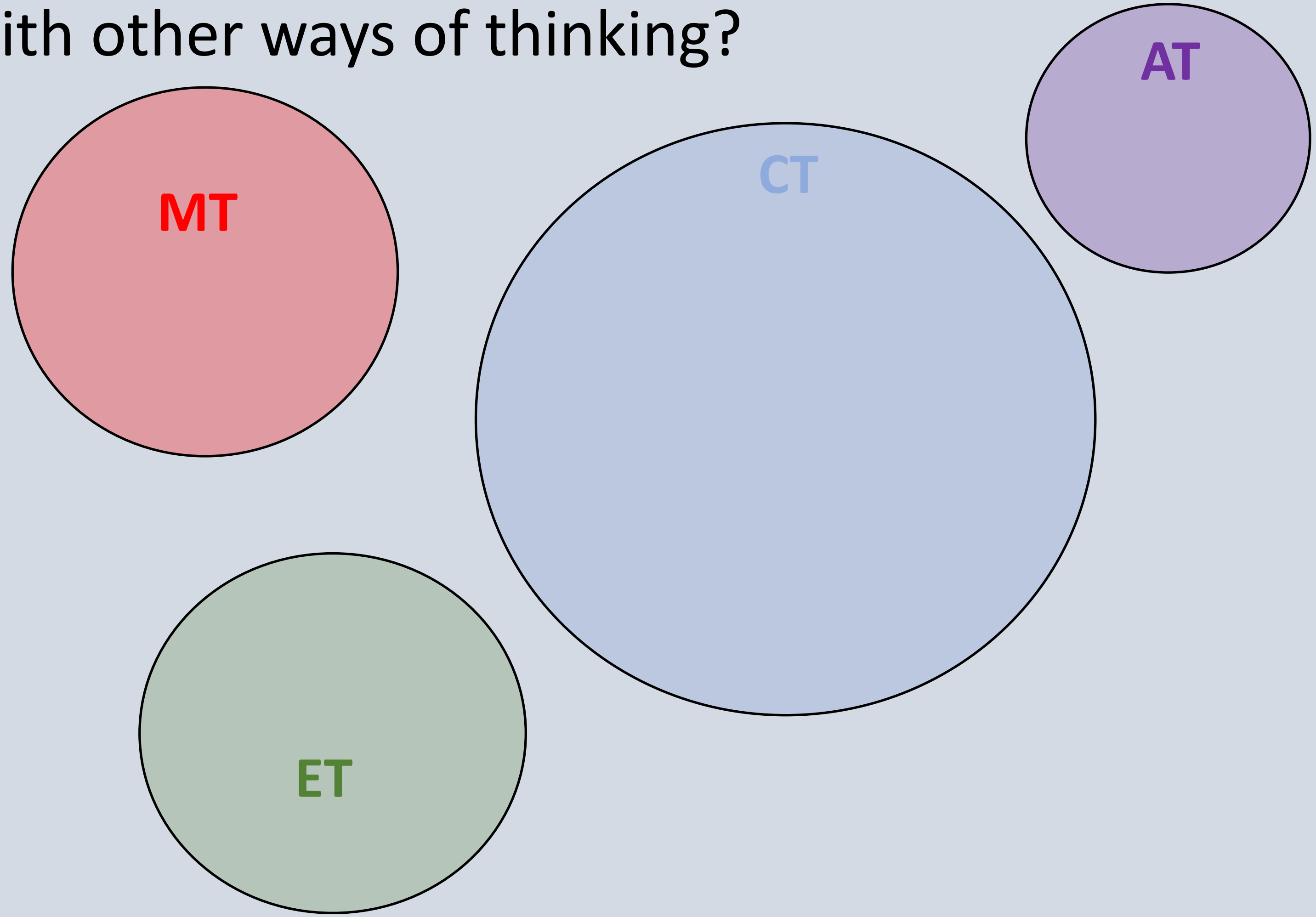


Taxonomy of Computational Thinking Practices for Mathematics and Science

(Weintrop, et al., 2016)

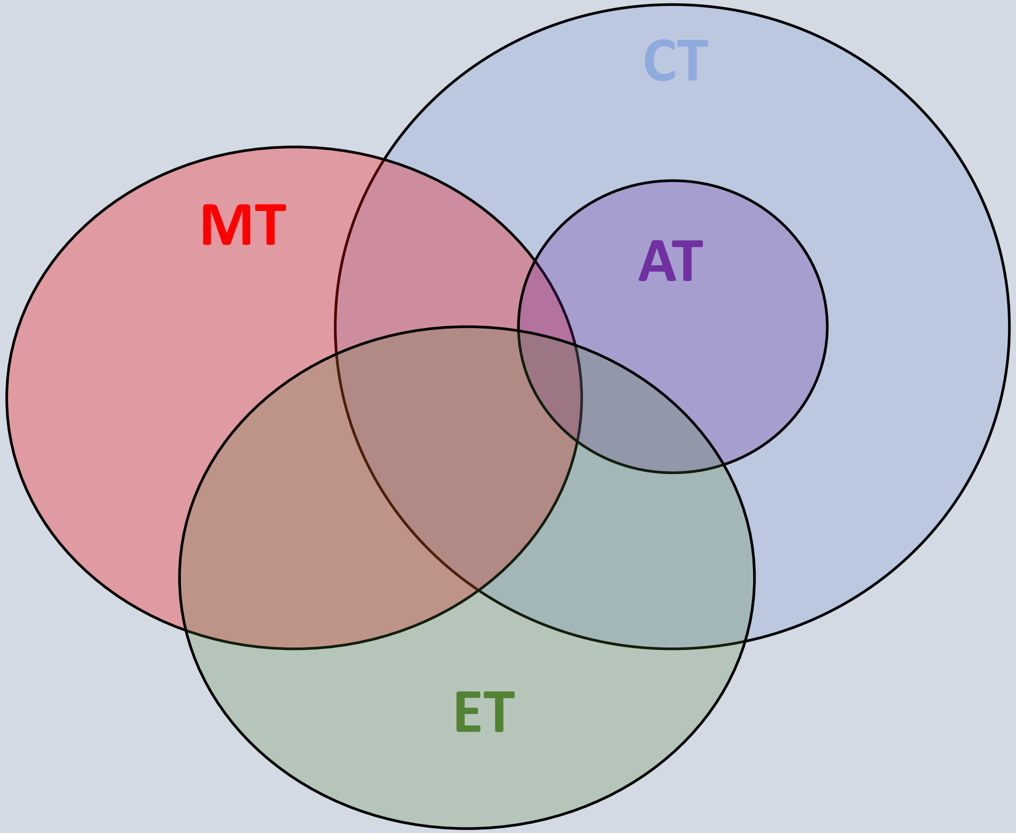
1. Theoretical investigations into the nature of CT itself

- How does CT interact with other ways of thinking?
  - Mathematical Thinking
  - Engineering Thinking
  - Algorithmic Thinking



1. Theoretical investigations into the nature of CT itself

- How does CT interact with other ways of thinking?
  - Mathematical Thinking
  - Engineering Thinking
  - Algorithmic Thinking



1. Theoretical investigations into the nature of CT itself

- Open Research Questions:

- What is CT?
- How might CT be taught, learned, and assessed?
- What “counts” as CT and what doesn’t? Can there be computation without CT?
- Is there CT for mathematics? For physics? For biology?

1. Theoretical investigations into the nature of CT itself

- Possible Research Methods:

- Synthesize the literature to compare and contrast current perspectives
- Interview experts in different STEM disciplines about computation and CT in their fields
- Design (and use) assessments for measuring and evaluating CT

2. How does CT help students learn mathematical or scientific content?

- Example from my grant: “Developing Undergraduate Combinatorial Curriculum In Computational Settings”
- In combinatorics, it is important for students to know when to add versus when to multiply when they solve counting problems
- How does writing a program help a student better understand when to multiply or add in counting?

2. How does CT help students learn mathematical or scientific content?

- Suppose there are four different French books, three different Russian books, and five different Spanish books. In how many ways can you take two of those books with you, if the two books are not in the same language?

FR FS RS

$$4x3 + 4x5 + 3x5$$

- I hypothesize that the act of programming will help students better understand when they should multiply or add in a counting problem

2. How does CT help students learn mathematical or scientific content?

```
F = ['F1','F2','F3','F4']
R = ['R1','R2','R3']
S = ['S1','S2','S3','S4','S5']

for i in F:
   for j in R:
      print(i,j)
for i in F:
   for j in S:
      print(i,j)
for i in R:
   for j in S:
      print(i,j)
```

```
F1 R1
F1 R2
F1 R3
F2 R1
F2 R2
F2 R3
F3 R1
F3 R2
F3 R3
F4 R1
F4 R2
F4 R3
```

```
F1 S1
F1 S2
F1 S3
F1 S4
F1 S5
F2 S1
F2 S2
F2 S3
F2 S4
F2 S5
F3 S1
F3 S2
F3 S3
F3 S4
F3 S5
F4 S1
F4 S2
F4 S3
F4 S4
F4 S5
```

```
R1 S1
R1 S2
R1 S3
R1 S4
R1 S5
R2 S1
R2 S2
R2 S3
R2 S4
R2 S5
R3 S1
R3 S2
R3 S3
R3 S4
R3 S5
```

$4x3 + 4x5 + 3x5 = 47$

2. How does CT help students learn mathematical or scientific content?

- Open Research Questions:

- How does engaging in a computational thinking and activity affect students’ understanding of a given mathematical or scientific concept
- How much (and what kinds of) computation is necessary to see gains in student understanding?
- How can computation practically be implemented into undergraduate and graduate classrooms?

2. How does CT help students learn mathematical or scientific content?

- Possible Research Methods:

- Interview students to gain in-depth insight into how they leverage CT to conceptualize ideas in science and mathematics
- Run comparative studies between students or classrooms in which computational thinking and activity is a variable
- Develop research-based curricula that leverage computational thinking and activity to learn concepts

3. How does CT help students develop mathematical or scientific practices?

- By practices I mean broad, overarching activities or perspectives that we want to foster in students
- Mathematical practices include problem solving, generalizing, justifying, proving, etc.

- Here “practices” could also entail dispositions or beliefs
- The ability to recover from mistakes, perseverance, beliefs about oneself, identity as a scientist or mathematician, etc.

3. How does CT help students develop mathematical or scientific practices?

- Open Research Questions:

- How can CT help to develop desirable dispositions and attitudes?
- In what ways does the debugging process during programing affect students’ ability to recover from mistakes?
- Do students who engage in computational thinking and activity develop better and longer lasting problem solving skills?

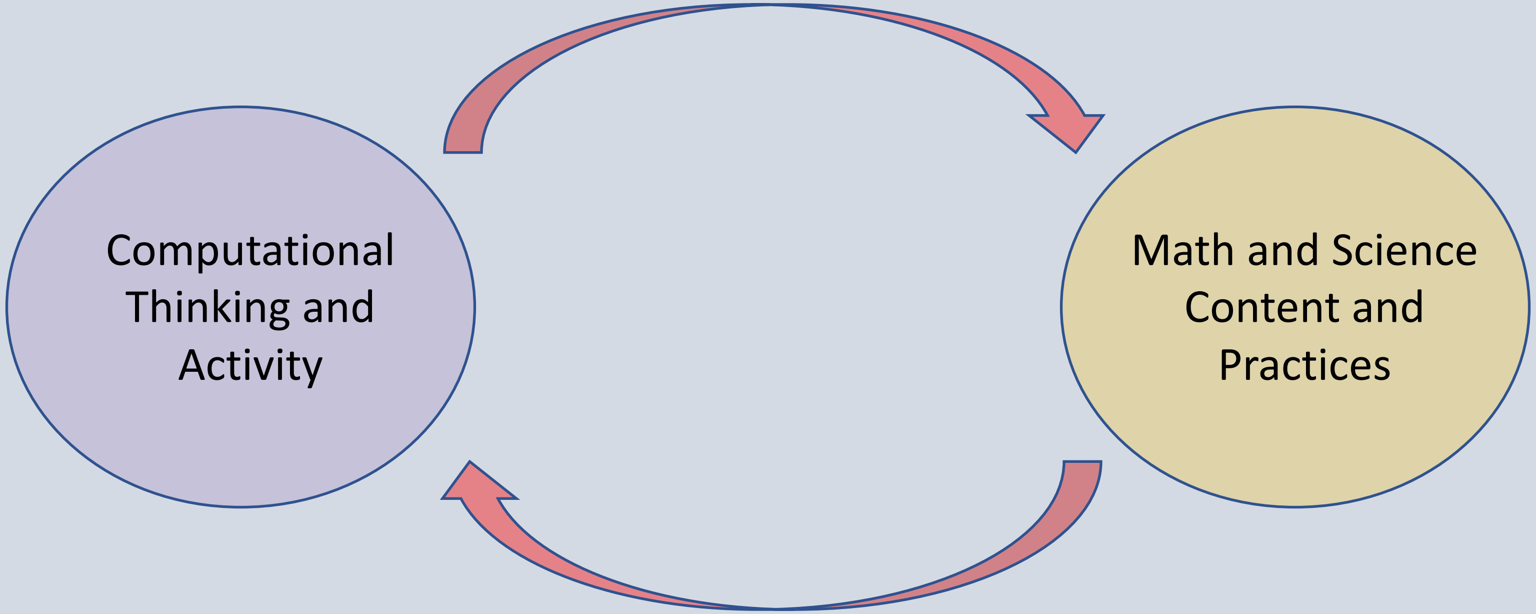
3. How does CT help students develop mathematical or scientific practices?

- Possible Research Methods:

- Interview experts about their learning of practices and how they view computational thinking and activity as solidifying their practices
- Conduct interviews with students about their beliefs about science before and after they engage in computational thinking and activity
- Conduct longitudinal studies with groups of students to investigate the long-term effects of computational thinking and activity on their problem solving practices

An Additional Note

- So far I have talked about using CT to learn content and practices
- But, we can also study how we can use content and practices to help students develop computational thinking and activity



Conclusion

- There are a variety of definitions and characterizations of computational thinking, but “it is time to redress the gaps and broaden the 21st-century academic discourse on computational thinking” (Grover & Pea, 2013)
- Educational research affords many methods and approaches we could use to study these questions and phenomena
- The time is ripe to engage exciting research related to computational thinking!

Thank you!

Elise.Lockwood@oregonstate.edu

Combinatorialthinking.com