NICE Webinar Series

NATIONAL INITIATIVE FOR CYBERSECURITY EDUCATION

Computational Literacy - A New Literacy Necessary for the Future of Learning and Work

February 16, 2022

Computational Literacy

The Very Idea

Andrea A. diSessa
Graduate School of Education
UC Berkeley

Exploring the “big picture” of what we should be doing concerning computation and education.

Outline

1. The Literacy Model

    Computation as a literacy

    – Historical View of the development of literacies, both textual and technical.

    – Principles for the Literacy Model

2. Real-world vignettes of computational literacy in the classroom.

    – More Principles for the Literacy Model

3. Critique of an alternate model

    Computational Thinking

A Historical Example:
Galileo’s Tribulations

A personal puzzle about: *Dialogues Concerning Two New Sciences*

Six theorems and their proofs

Think: What is going on here?

- What’s the topic?
- Why is Galileo “beating around the bush”?

Six Theorems

If a moving particle, carried uniformly at constant speed, traverses two distances, then the time intervals required are to each other in the ratio of these distances.

Theorem 2

Theorem 3

Theorem 4

Theorem 5

Theorem 6

Six Theorems

Theorem 1

If a moving particle traverses two distances in equal intervals of time, these distances will bear to each other the same ratio as their speeds. And conversely, if the distances are as the speeds, then the times are equal.

Theorem 3

Theorem 4

Theorem 5

Theorem 6

Six Theorems

Theorem 1

Theorem 2

Theorem 3

If two particles are carried with uniform motion, but each with a different speed, then the distance covered by them during unequal intervals of time bear to each other the compound ratio of the speeds and time intervals.'

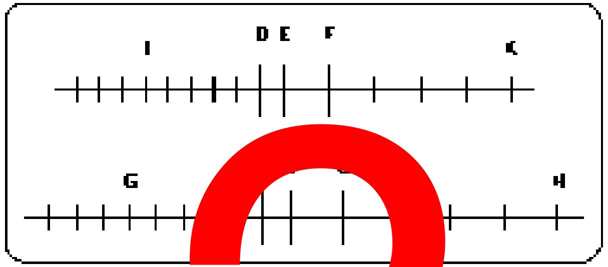
Theorem 5

Theorem 6

Proof of Theorem 1

If a moving particle, carried uniformly at constant speed, traverses two distances, then the time intervals required are to each other in the ratio of these distances.

Let a particle move uniformly at a constant speed through two distances AB, BC, and let the time required to traverse AB be represented by DE; the time acquired to traverse BC, by EF;



then I say that the distance AB is to the distance BC as the time DE is to the time EF.

Let the distances and times be extended on both sides towards G, H and I, K; let AG be divided into any number whatever of spaces each equal to AB, and in like manner lay off in DI exactly the same number of time-intervals each equal to DE. Again lay off in CH any number whatever of distances each equal to BC; and in FK exactly the same number of time-intervals each equal to EF; then will the distance BG and the time EI be equal and arbitrary multiples of the distance BA and the time ED; and likewise the distance HB and the time KE are equal an arbitrary multiples of the distance CB and the time FE.

And since DE is the time required to traverseAB, and the time EI will be required for the whole distance BG, and when the motion is uniform there will be in EI as many time intervals each equal to DE, as there are distances in BG each equal to BA; and likewise it follows that KE represents the time required to traverse HB.

Since, however, the motion is uniform, it follows that if the distance GB is equal to the distance BH, then must also the time IE the equal to the time EK; and if GB is greater than BH, then also IE will be greater than EK; and if less, less. There are then four quantities, the first AB, the second BC, the third DE, and the fourth EF; the time IE and the distance GB are arbitrary multiples of the first and the third, namely of the distance AB and the time DE.

But it has been proved that both of these latter quantities are are either equal to, greater than, or less than the time EK and the space BH, which are arbitrary multiples of the second and the fourth. Therefore the first is to the second, namely the distance AB is to the distance BC, as the third is to the fourth,namely the time DE is to the time EF.

Q. E. D.

The Solution!

$d = rt$

BUT:

Galileo did not know algebra!

It did not exist when he did his work !

A New Representational System

Algebra!

$$d = r\,t \quad \longrightarrow \quad \frac{d_1}{d_2} = \frac{r_1}{r_2}\,\frac{t_1}{t_2}$$

Theorem 1

Theorems 2-6

A New Representational System

Algebra!

$$d = r\,t \quad \longrightarrow \quad \frac{d_1}{d_2} = \frac{r_1}{r_2}\,\frac{t_1}{t_2}$$

In the case that $r_1 = r_2$, the r terms cancel,

leaving $\dfrac{d_1}{d_2} = \dfrac{t_1}{t_2}$.

Theorems 2-6

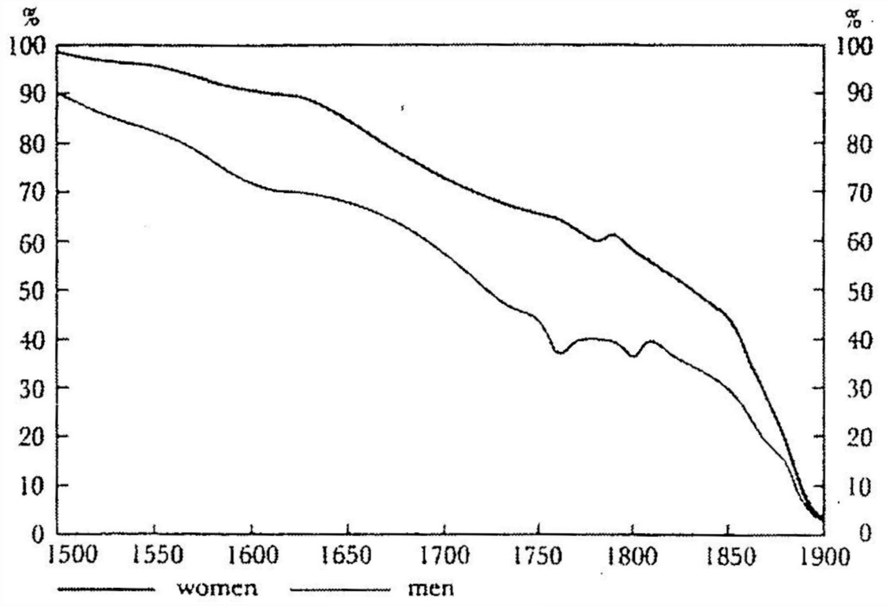
Notes on the Development of Algebra as a Technical Mass Literacy

- Took 300 years (more or less)!
- 1900: In the first decade of the 20 th Century: ~ 60% of high school students
- 1925: A severe critique on importance and learnability of algebra (William Heard Kilpatrick)
- 1955: ~25% !
- 2013: ~80% or above

Equity in the Development of Algebra as a Mass Literacy

- "Recent data" from NCES
  Categories: Whites, Blacks, Asian
- Basic Algebra
  - Almost no differences; Blacks lead in some categories! (Thanks, Bob Moses!)
- Calculus
  - Asian: 43%
  - White: 16%
  - Black: 6%

World, English, and US Illiteracy Rates, 1500 -1900



Literacy for selected areas in Europe and North America c. 1700

Region	Male literacy rate	Female literacy rate
England	40%	25%
New England	70%	

Principles for Computational Literacies

1. The development of a literacy is a very complex and extended social & cultural change.
2. A literacy fundamentally alters the intellectual structure of domains.
   A literacy needs a literature
   You don’t get “smart” by learning to read. You get “smart” reading Shakespeare or a math book.
3. Re-Mediation is a primary mode of change.

QUESTIONS?

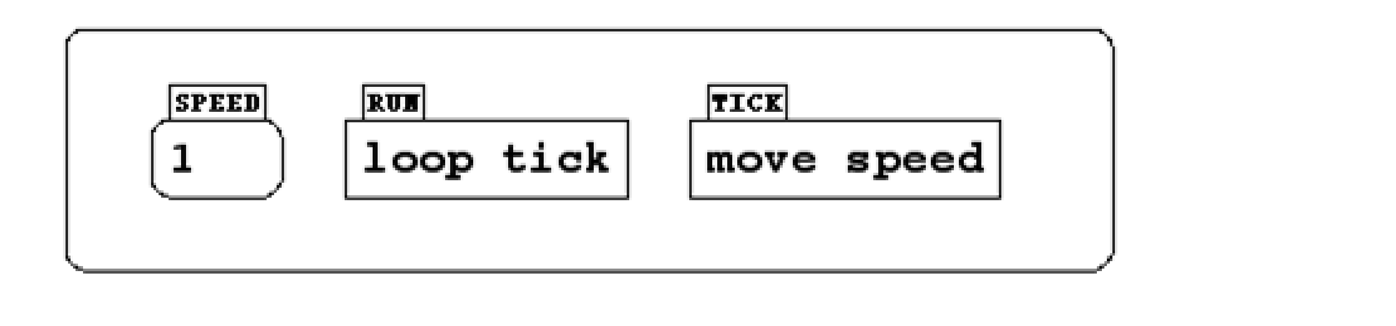
Green Shoots of a Computational Literacy: Motion

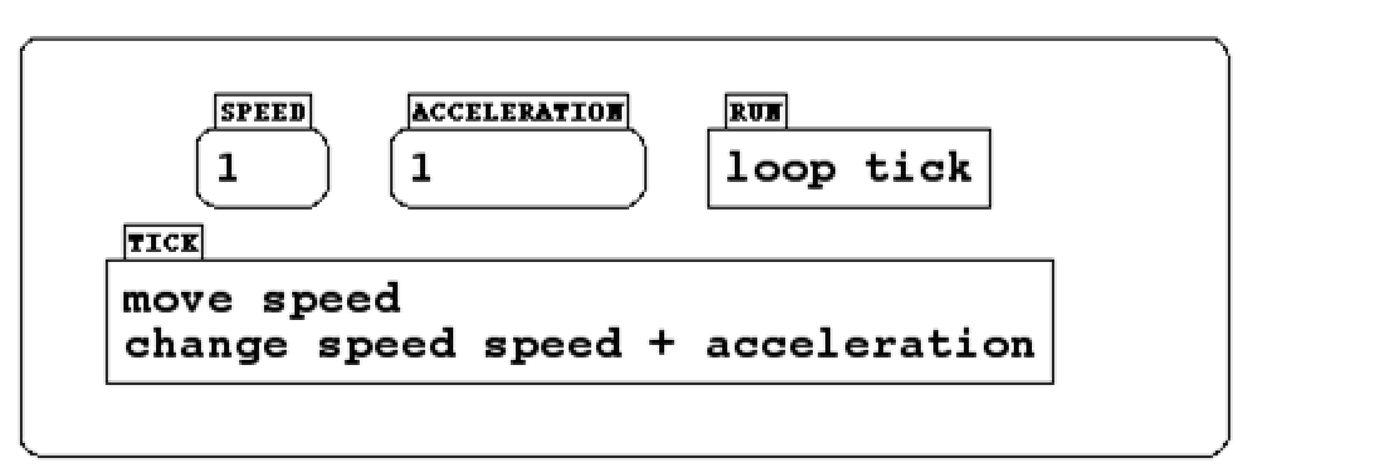
Teaching about motion (high school “physics”) to sixth graders. A year-long class, without algebra and calculus!

Topics: Kinematics (the mathematics of motion) incl. vectors, graphing (representations of motion), composing motions, frames of reference, $F = ma$

Defining Speed and Acceleration: The Tick Model

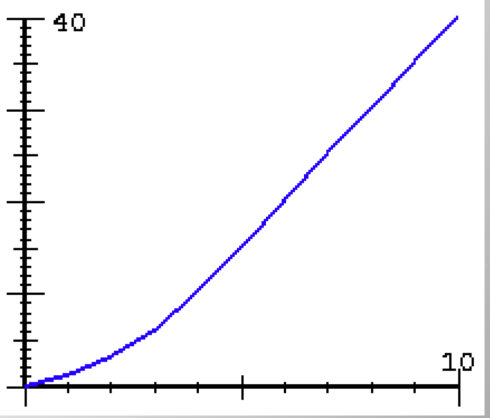
$$x = vt \qquad v = \frac{dx}{dt} \qquad a = \frac{dv}{dt} \qquad x = \int v\, dt \qquad v = \int a\, dt$$



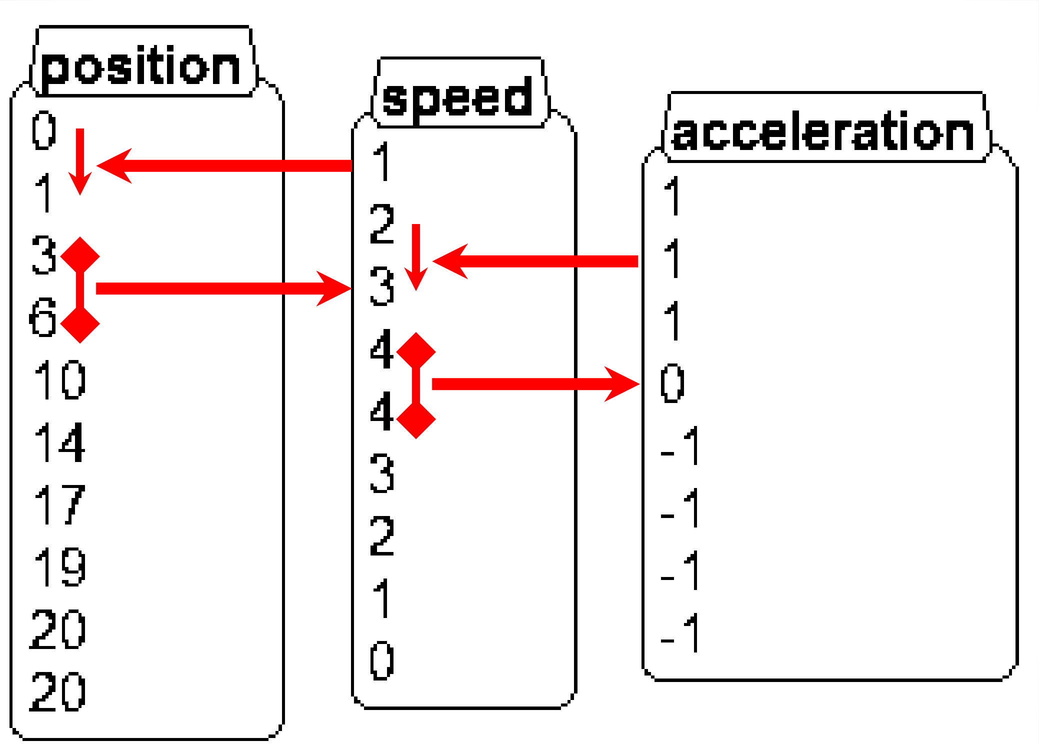


Advantages of the Tick Model

- No prerequisites (algebra)
- Cognitive simplicities (discrete time; natural numbers)
- Phenomenological: Makes motion!
- Generative model for any dynamical simulation
- Basis for projects (games!)
- Conceptual foundation for calculus
  - Reading speed off position graphs
  - Amy's Innovation
  - Charlie's Insight



Our Extension of Charlie's Insight



Number lists, rather than algebraic expressions, as the core representation of functions.

Students can invent and enact the fundamental theorem of calculus: derivative and integral are inverses.

Recall the Dispiriting Demographics on Calculus

If children can do this in the sixth grade, what will they be capable of doing in high school?

Might this approach REPLACE algebraic calculus for many learning paths?

Principles for Computational Literacies

4. Reformulation

“Redesigning knowledge” based on “cognitive simplicities”

“Speed is NOT the ratio of infinitesimals, but a small number that plays a very particular role in a simple program that moves objects around.”

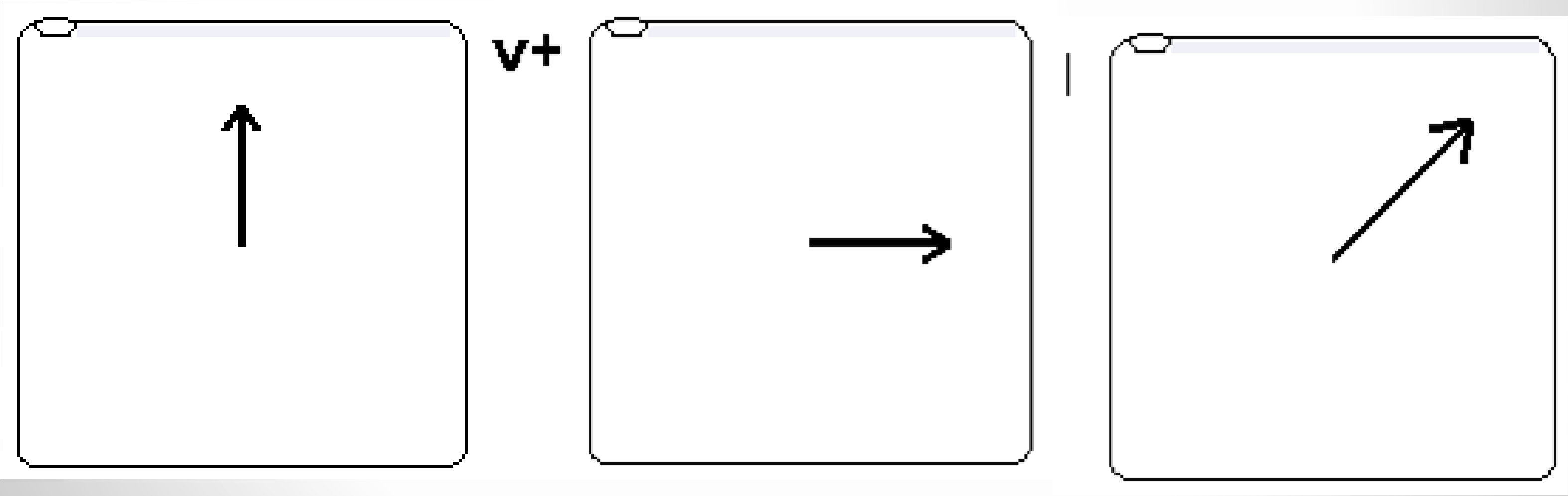
A synergy of Re-Mediation + Reformulation is especially powerful!

Vectors

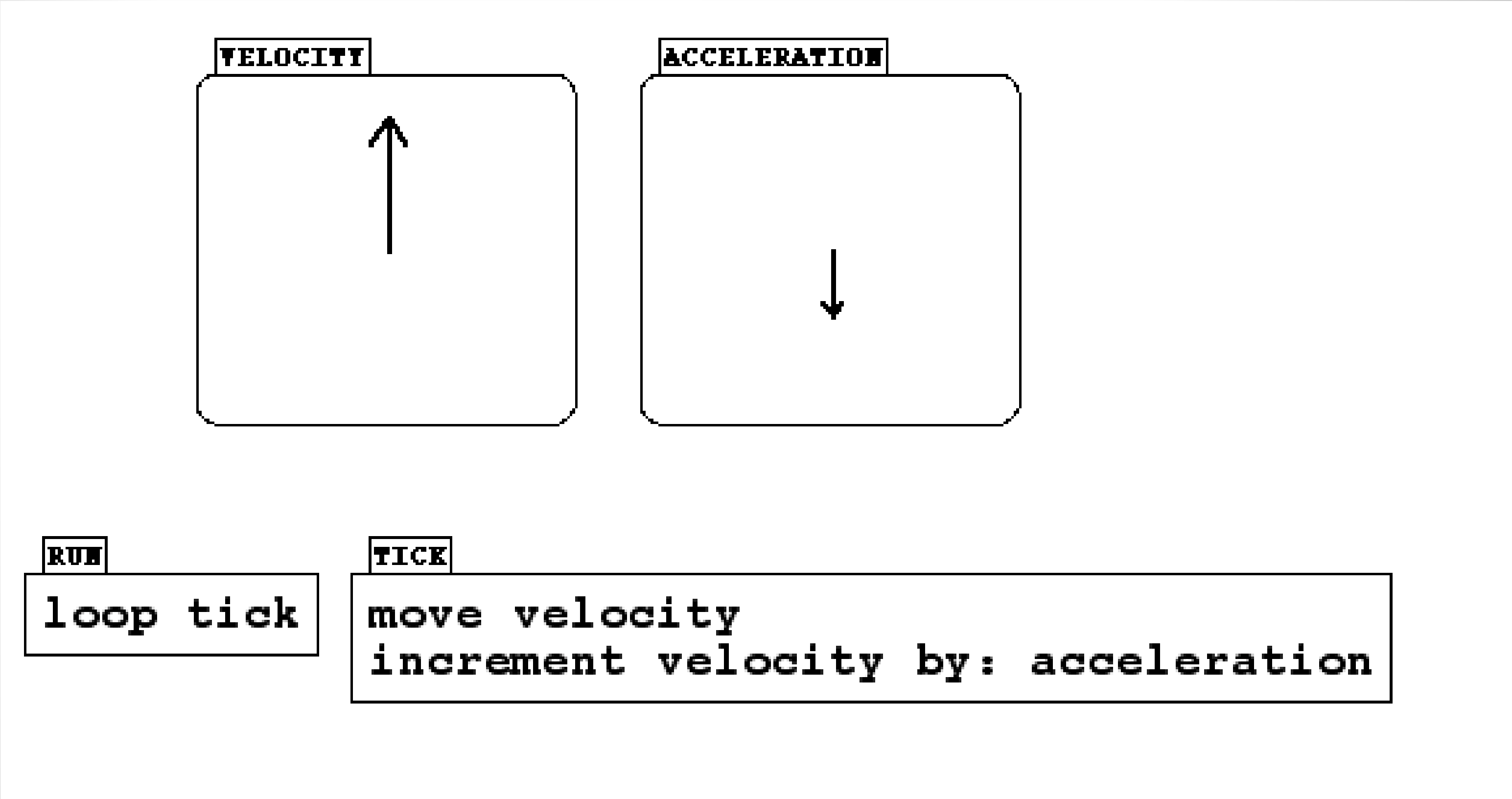
- The NSF said: “Vectors are simply developmentally inappropriate .” Reject!
- But all of the conceptual and pragmatic advantages of discrete time, functions -as-lists, etc. apply!

Vectors

(Bruce Sherin)



The Vector Tick Model



What Happened with Vectors:

- Students learned about vectors easily.
- Massive time and experience with them.
  - Students LOVED playing with vectors.
  - They used vectors as user interface devices for computer games they invented and wrote.
- Magic in a nutshell: Phenomenologically live, useful, + time

Principles for Computational Literacies

5. Revitalizing the ecology of learning activities (for "substance" and engagement!)

- Reduce or eliminate small, algebra-based problems
- Much greater emphasis on phenomenology and explanation
- More extended, self-driven engagement: Projects such as writing computer games
- Much greater opportunity for personalization (including personal intellectual excursions such as Amy's or Charlie's)

QUESTIONS?

Computational Thinking

A HUGE deal; a major movement

- $100 million funding for NSF.
- Arguably catalyzed a national curriculum in the UK.

(Typically, the educational story is much more complex in the U.S.)

Computational Thinking

Jeannette Wing: 2006, 2008, 2011, 2014

Wing's notion of Computational Thinking:

- … "is thinking like a computer scientist"
- … "involves solving problems, designing systems, and understanding human behavior, by *drawing on the concepts fundamental to computer science*"

Wing's Appeal to "Higher Order Thinking Skills" (HOTS)

"[Computational thinking] represents a *universally applicable attitude and skill set everyone, not just computer scientists, would be eager to learn and use.*"

(a) Divide and Conquer
(b) Planning
(c) Design a Representation
(d) Abstraction

Précis of the Knotty History of HOTS

1. Plato (~400 B.C.):

   "Those who are by nature good at calculation are, as one might say, *naturally sharp in every other study*, and … those who are slow at it, *if they are educated and exercised in this study*, nevertheless *improve and become sharper* than they were."

Précis of the Knotty History of HOTS

2. “Mental Discipline” and “faculty psychology” (~1900 -30)

A diverse but powerful set of general skills (quickness, attention, discrimination, …) can be improved by immersion in a selected set of disciplines (mathematics, classical studies such as Latin …).

Edward Thorndike

– Credited with a “crushing blow” to these
– Massive empirical study circa 1925

Précis of the Knotty History of HOTS

Thorndike (1924)

"The intellectual values of studies ["disciplines"] should be determined largely by the *special information, habits, interests, attitudes, and ideals* which they demonstrably produce. The expectation of any large differences in general improvement of the mind from one study ["discipline"] rather than another seems doomed to disappointment."

Précis of the Knotty History of HOTS

Thorndike and Woodworth (1901)

“It is misleading to speak of sense discrimination, attention, memory, observation, accuracy, quickness, etc.” This is so because, “Multitudinous separate individual functions are referred to by any one of these words. These functions may have little in common.”

Multiplicity

Précis of the Knotty History of HOTS

3. George Pólya and Problem Solving – 1945, then a BIG boom in the 1970s & 80s

(a) Divide and Conquer
(b) Planning
(c) Design a Representation ("Draw a Picture," The Importance of Notations)
(d) Abstraction (Generalization)

Wing's Appeal to "Higher Order Thinking Skills" (HOTS)

"[Computational thinking] represents a *universally applicable attitude and skill set everyone, not just computer scientists, would be eager to learn and use*"

(a) Divide and Conquer
(b) Planning
(c) Design a Representation
(d) Abstraction

Précis of the Knotty History of HOTS

Andy's pithy summary of the Problem Solving Movement: Basically, it failed.

- Problem solving has not come to a dominant presence in any discipline, much less in all.
- The best way to learn to solve problems (e.g., in mathematics) is to master the concepts.
- Learning problem solving is much more difficult than you might think because of the "multiplicity problem ." The "same thing" appears to have very many different forms.

Modern HOTS (Meta -CS)

Reflection: Questions for Wing

- What makes your list different/better than Pólya’s?
- Why do you think the program will work better now than before?
- What is your response to the “ multiplicity problem”?
- What makes you think CS can wrest “abstraction” away from other disciplines, like mathematics?

HOTS Critique: Summary

- HOTS is an ancient, persistent, and psychologically powerful meme (perhaps impervious to the results of scientific scrutiny) that provides a central pillar of face value to Computational Thinking.
- But the history of HOTS is complex and problematic, and that makes it an insecure foundation for CT.

Review and Reflection: Principles

1. The growth of a literacy is a complex and very extended social and cultural process.
2. Re-organizing intellectual domains: Deep shifts in the cognitive/conceptual structure of domains, when and how they can be taught.
3. Re-Mediation: Computation's properties as a representational system will be foundational to the transformation.

Review and Reflection: Principles

4. Reformulation: In addition to re-mediation, we may often be able to engage deep "cognitive simplicities." Reformulation + Re-Mediation can be an especially powerful synergy

5. Revising, revitalizing the fabric of learning activities: Phenomenological engagement; projects and extended personal enterprises (among other modes) will become much more prevalent.

Personal Reflection

- No one can design or specify exactly what a computational literacy will eventually entail.
- Just get used to the fact that this is a “long - haul” project.

Final Words

Computational Literacy represents a powerful take on “the big picture” of computation’s potential effects on education, which has both scientific and practical advantages over other views. Keeping an eye on THAT big picture can be highly consequential over the long term.

- Doing things that have true cumulative power
- Monitoring the state of our accomplishments
- “Filtering” out “noise” and distractions from the big picture


THE END

Want to Know More?

- diSessa, A. A. (2000). *Changing Minds: Computers, Learning, and Literacy*. MIT Press.
- Contact: disessa@berkeley.edu – I can send you a paper expanding on this talk.
- The diSessa Foundation - A place to promote, collect, and coordinate work on the development of computational literacies. (Andy's textbook on a computationally literate physics.) Computational-literacy.org -- Stay Tuned.

Q & A

https://www.surveymonkey.com/r/febnicewebinar

Thank You for Joining Us!

Upcoming Webinar:
"The NICE Framework at Work - Use Cases from Industry"

When:
March 16, 2022, at 2:00-2:45PM ET

Register:
https://go.usa.gov/xteQd

nist.gov/nice/webinars