RCPLI Computational Thinking lesson, 3/11.

First: there are two things I wanted to include, but didn't have time for.

1) I wanted to start with the Math Salute, which I learned from James Tanton, but I couldn't afford the time. It's brilliant, and you can still get the experience of it by video. Here's the thing: you *must try it* in order to understand what's so funny about it. Math is not a spectator sport. Also, pause the (first) video at 1:45 to give your brain a little time to chew on it.

http://www.jamestanton.com/?p=1328

2) I wanted to finish with the Towers of Hanoi, which is a solitaire game played with a bunch of disks and three pegs. Play it here:

http://www.coolmath-games.com/0-tower-of-hanoi

and try to figure out the smallest number of moves that can solve the puzzle if there are n disks. I wanted to work toward the "recursive solution" to the puzzle. Recursion is an excellent example of computational thinking (and Math Practice 8). As for the solution: ask me later. It's really elegant, by which I mean it's unexpectedly simple for how powerful it is. I would put the solution here, but I don't want to turn your brain off. I found the Wikipedia treatment of it to be a little confusing.

Supporting links:

ISTE + CSTA operational definition of computational thinking

https://csta.acm.org/Curriculum/sub/CurrFiles/CompThinkingFlyer.pdf

Numberphile video: Brown's Criterion (binary magic trick)

https://www.youtube.com/watch?v=kQZmZRE0cQY

Lesson source:

csunplugged.org

Other worthwhile links:

Interesting TED talk by Ashley Gavin of Wesleyan University, Girls Who Code, etc.

https://www.youtube.com/watch?v=5jmN_tBS0t4

Bunch of free online lessons and such. These are the Hour of Code people.

https://code.org/

What I actually did in the lesson:

1) Info intended for professionals, not K-12 students—such as the ISTE/CSTA handout linked above, plus quotes listed on last page of this file.

2) Magic trick. Use number cards from BinaryMagicTrick.pdf. Email brad@humboldt.edu for Excel source if you want to modify it. See Numberphile video (link above) for the general plan…but the trick comes in two flavors:

a. Player silently picks a number, 1-31. For each sheet, player says yes or no. Magician has kept a running total of the top left corner numbers, and reports their sum.
b. Magician gives the sheets to the player, who announces a number (1-31) out loud. Magician says which sheets have player's number; they are the sheets corresponding to the 1's in the binary representation of the number. For instance,

   22 = 1(16) + 0(8) + 1(4) + 1(2) + 0(1). 22 appears on cards 16, 4, and 2. (If shown in the order 16-1, these cards are first, third, and fourth. If shown in the order 1-16, these cards are second, third, and fifth.

3) First activity (p.3) from csunplugged book, link above. Specifically, worksheets on p.6 and p.8. Use large dot cards from BinaryMagicTrick.pdf, small dot cards from csunplugged book, or paper-saving small cards from CardDots.PNG (adapted from csunplugged book).

4) Explain magic trick.

Computational Thinking Quotes

Jeanette Wing (researcher at Microsoft; previously a CS professor at USC, MIT, and Carnegie Mellon; also NSF Assistant Director for the Computer and Information Science and Engineering (CISE) Directorate):

"Computational thinking is a **fundamental skill** used by everyone in the world, and should be incorporated into educational programs along with reading, writing and arithmetic to grow every child's analytical ability."

(http://www.cs.cmu.edu/news/computational-thinking-should-be-honed-fundamental-skill-educational-programs-says-expert-cmu)

…and to clarify that quote further,

"A **fundamental skill** is something every human being must know to function in modern society."
https://www.cs.cmu.edu/~15110-s13/Wing06-ct.pdf

"Computational thinking is the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent."

and

"At Carnegie Mellon, computational thinking is everywhere. We have degree programs, minors, or tracks in "computational X" where X is applied mathematics, biology, chemistry, design, economics, finance, linguistics, mechanics, neuroscience, physics and statistical learning. We even have a course in computational photography. We have programs in computer music, and in computation, organizations and society."

http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why

Edsger Dijkstra (winner of the ACM Turing Award and, according to Wikipedia, "one of the most influential members of computing science's founding generation"):

"I don't need to waste my time with a computer just because I am a computer scientist."

and, attributed sometimes to Dijkstra, sometimes to others:

"Computer science is no more about computers than astronomy is about telescopes."

http://csunplugged.org/