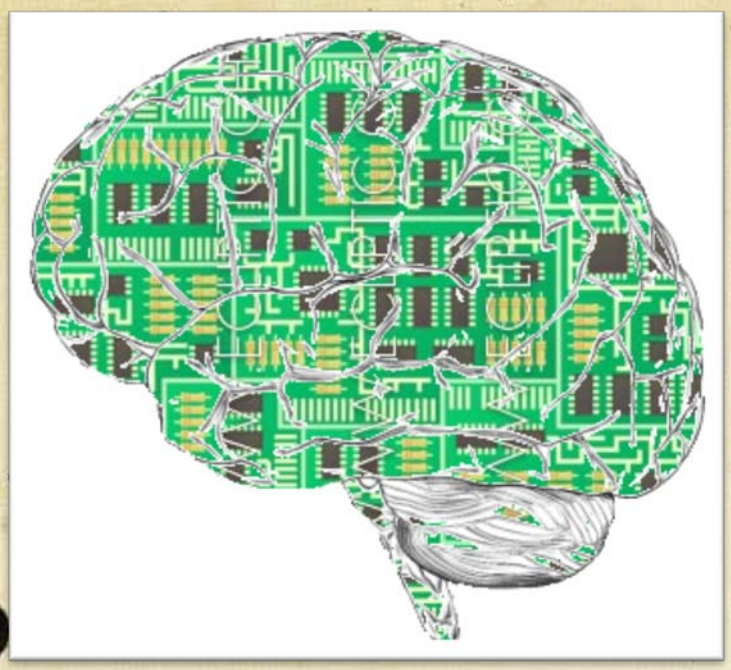
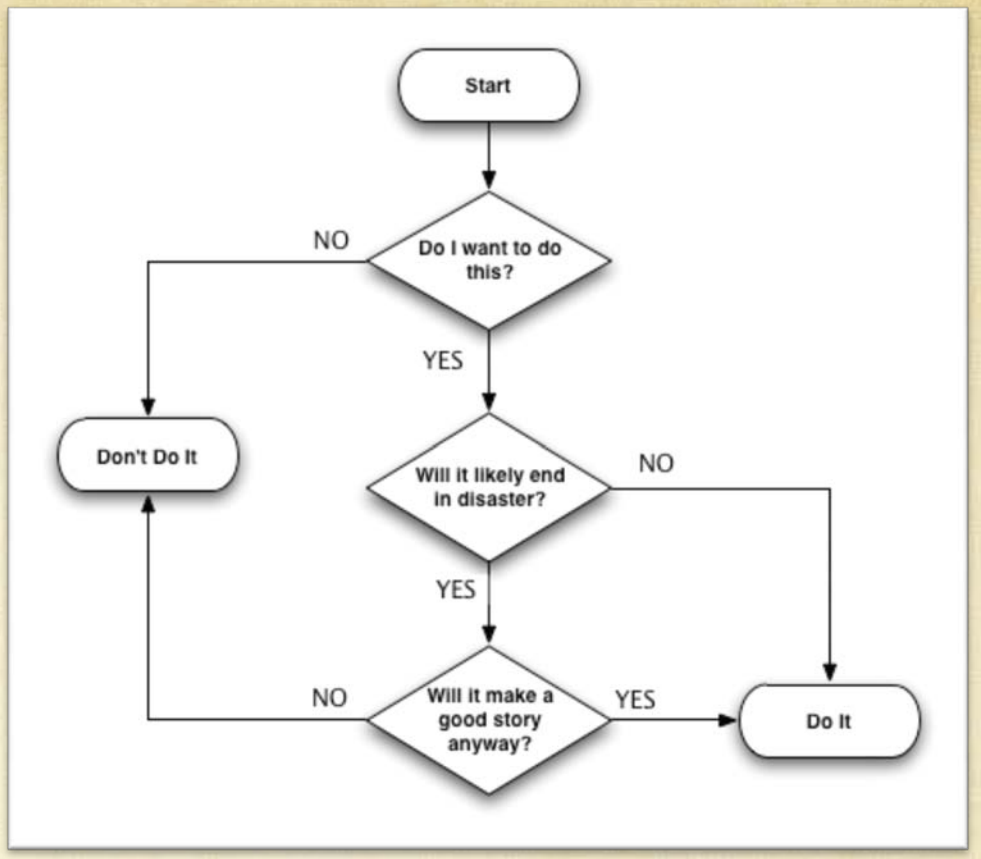




Computational Thinking and 21st Century Problem Solving

Dr. Aman Yadav
February 9, 2011

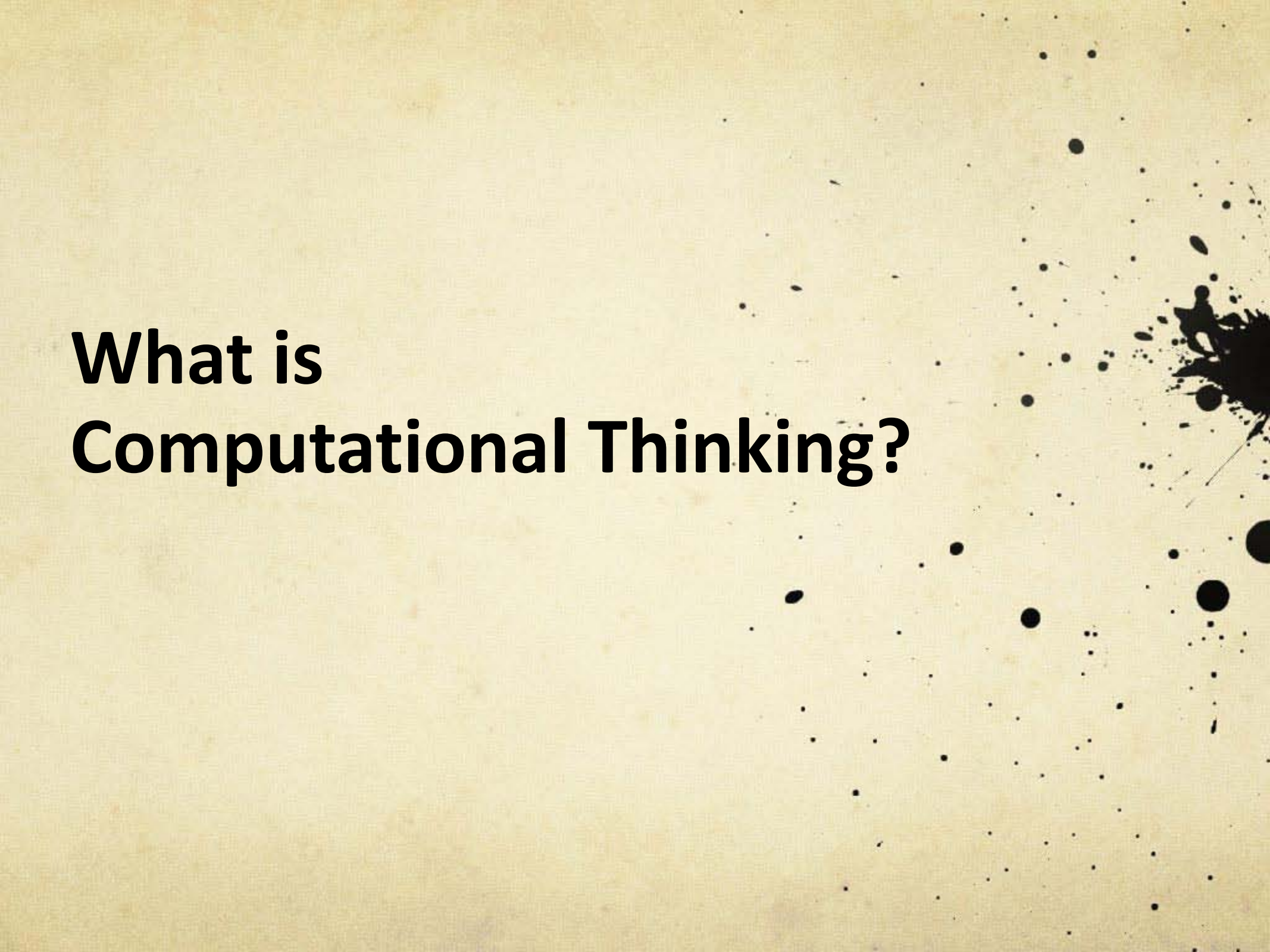
PURDUE
UNIVERSITY

Outline

- What is Computational Thinking?
  - Definition and Vision
  - Common Examples

- Computational Thinking Concepts
  - Abstraction
  - Logical thinking
  - Algorithms
  - Debugging

What is Computational Thinking?



Example: Driving Directions

How do you give driving directions from Purdue to the mall?

Clicker Question 1

How extensive were your directions?

[A] One step (e.g., type “Tippecanoe mall" into GPS / Google)

[B] Two steps (e.g., from downtown take CityBus 4A)

[C] Several steps (e.g., head East on 26 to 38 to 52)

[D] A detailed, turn-by-turn route (e.g., from BRNG)

[E] None of the above (e.g., "Man, I was way off!")

Clicker Question 2

How did you figure out the driving directions?

[A] Knew them already; simply "recalled" the route

[B] Sketched out a high-level map on paper

[C] Thought about several ways, picked one

[D] Texted a friend when no one was looking

[E] Modeled the entire Greater Lafayette Area as an undirected graph, solved the "single-pair shortest path problem," and applied it to the source and destination

What just happened? (Reflect…)

- How did you *think* about the problem?
  - Thought it through in your head - Mental Model
  - Developed step-by-step route - Algorithm
    - What's the best way to get there;
      reduce number of lights and turns - Efficiency

- Other Issues
  - What if scenarios – "Logical Thinking"
    - 26 was closed for construction
    - Home football game; need to avoid traffic
    - Time of day, weather, accidents, …
  - Abstraction – How to effectively give *any* directions
  - Automation – How to design a system like Google Maps

What is Computational Thinking?

- Definition
  - "CT involves solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science."
- Vision
  - A fundamental skill used by everyone by the middle of the 21st century (i.e., like reading, writing, and arithmetic).

J.M. Wing, "Computational Thinking," CACM viewpoint, vol. 49 no. 3, March 2006, pp. 33-35.

Clicker Question 3

What best describes the term "computer science?"

[A] Study of automatic computing

[B] Study of phenomena surrounding computers

[C] Study of what can be automated

[D] Study of computation

[E] Study of information processes

Clicker Question 3 (answer)

What best describes the term "computer science?"

[A] Study of automatic computing (1940s)

[B] Study of phenomena surrounding computers (1960s)

[C] Study of what can be automated (1970s)

[D] Study of computation (1980s)

[E] Study of information processes (2000s)

Denning, P. "Computing Field: Structure". In Wiley Encyclopedia of Computer Science and Engineering (B. Wah, Ed.). Wiley Interscience (2008).

Daily Examples of CT

- Looking up a name in an alphabetically sorted list (Binary Search)
  - e.g., 100 names per page in list of 150,000 names
  - How to minimize the number of pages to look at?

- You and your friend are buying tickets for a movie (Parallel Processing).
  - There are three independent lines
  - How do you get your tickets ASAP?

Clicker Question 4

What is the best way to serve 20 pizzas to 60 hungry students?
(i.e., How do you minimize the time for everyone to get pizza?)

[A] One table with all pizzas (the usual case)

[B] Five tables with four pizzas each

[C] People stay put and pizzas are passed around

[D] Four servers bring the pizza around to others

Clicker Question 4 (answer)

What is the best way to serve 20 pizzas to 60 hungry students?
(i.e., How do you minimize the time for everyone to get pizza?)

[A] One table with all pizzas (the usual case) – NO!

[B] Five tables with four pizzas each

[C] People stay put and pizzas are passed around

[D] Four servers bring the pizza around to others

Summary: Computational Thinking is…

- Conceptualizing, not programming
  - Not just technical details for using software
- Fundamental, not a rote skill
  - Not just one more thing to add to your curriculum
- A way that humans, not computers, think
  - Combines problem solving and critical thinking
- Ideas, not artifacts
  - How we use higher-level thinking to create solution
- For everyone, everywhere

Computational Thinking Concepts

Computational Thinking Concepts

1. Abstraction
2. Logical Thinking
3. Algorithms
4. Debugging

Concept: Abstraction

- Decomposition
  - "Computational thinking is reformulating a seemingly difficult problem into one we know how to solve."

- Abstraction
  - Pulling out the important details
  - Identifying principles that apply to other problems/situations

Clicker Question 5

Which of the following is NOT like the others?

[A] People standing in line at the store

[B] List of print jobs waiting to be printed

[C] Set of tennis balls in their container

[D] Vehicles lined up behind a toll booth

[E] Patients waiting to see the doctor

Clicker Question 5 (answer)

Which of the following is NOT like the others?

[A] People standing in line at the store (queue)

[B] List of print jobs waiting to be printed (queue)

[C] Set of tennis balls in their container (stack)

[D] Vehicles lined up behind a toll booth (queue)

[E] Patients waiting to see the doctor (queue)

Clicker Question 6

Which of the following is NOT like the others?

[A] Files and directories on a hard disk.

[B] Parents and children in a pedigree chart.

[C] Brackets in the NCAA basketball tournament.

[D] My closest friends on Facebook / Twitter.

[E] The format of XML or PDF documents.

Clicker Question 6 (answer)

Which of the following is NOT like the others?

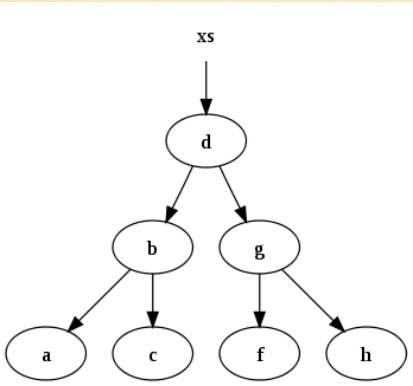
[A] Files and directories on a hard disk. (tree)

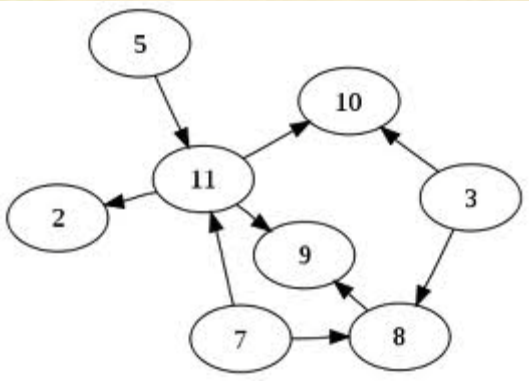
[B] Parents and children in a pedigree chart. (tree)

[C] Brackets in the NCAA basketball tournament. (tree)

[D] My closest friends on Facebook / Twitter. (graph)

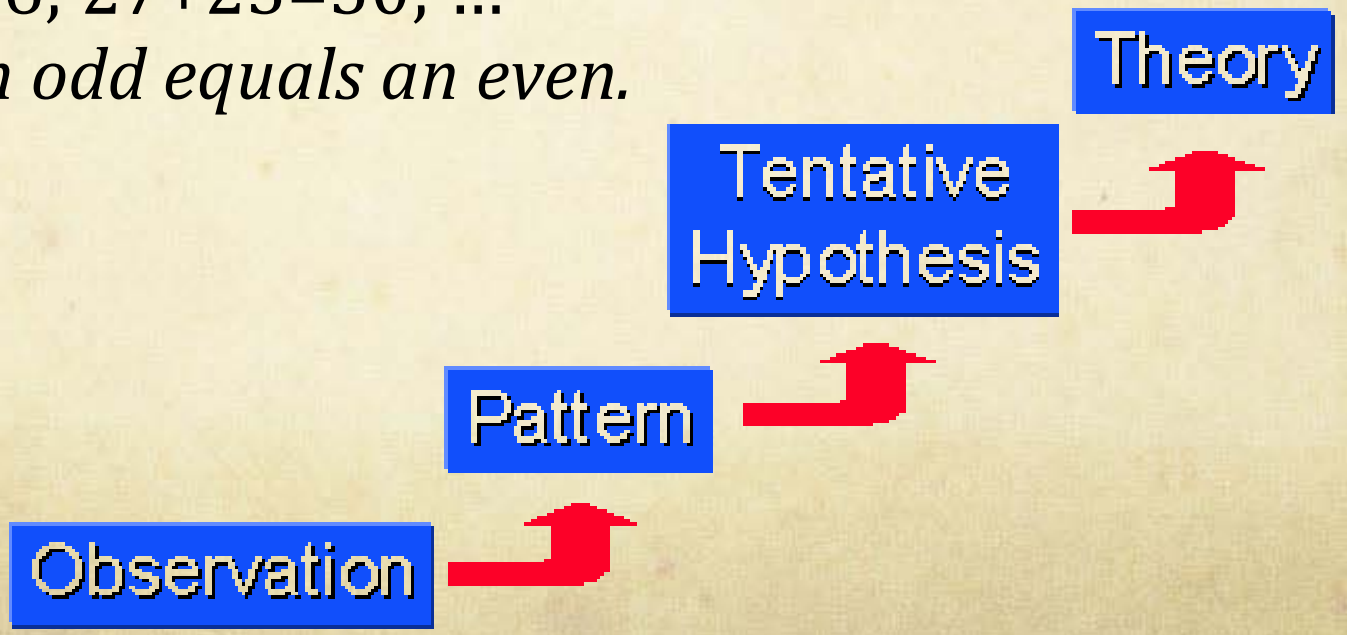
[E] The format of XML or PDF documents. (tree)





Concept: Logical Thinking

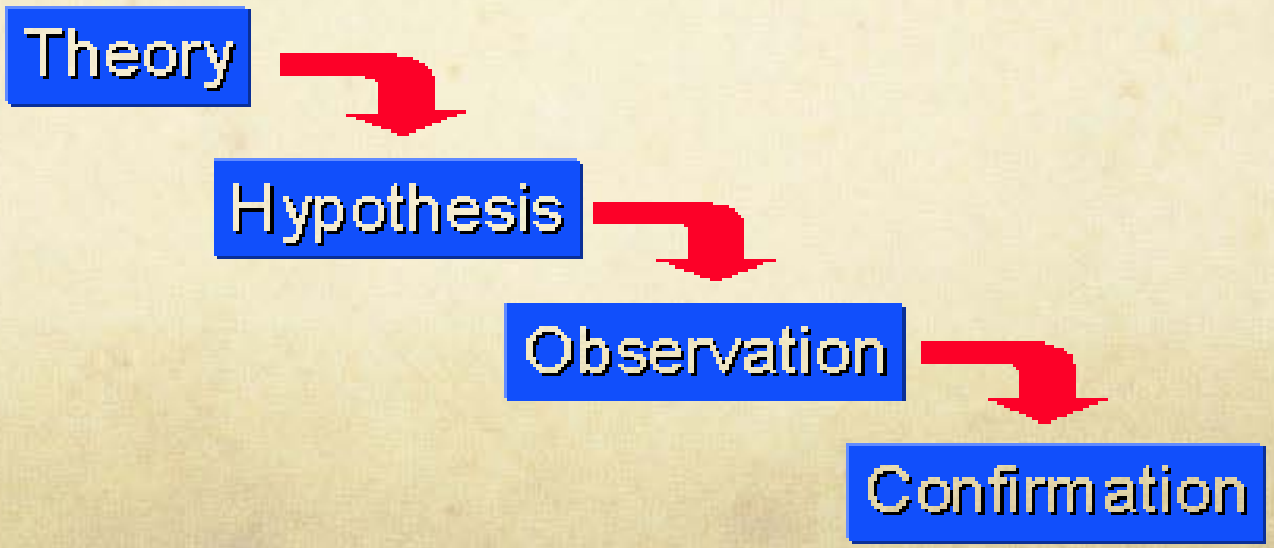
- Inductive reasoning
  - From specific examples to general principles.
  - Examples:
    - $8 \div 1 = 8$; $27 \div 1 = 27$; $118 \div 1 = 118$; …
      Dividing any number by 1 equals the number.
    - 3+5=8; 7+9=16; 27+23=50; …
      An *odd plus an odd equals an even.*



Concept: Logical Thinking

- Deductive reasoning
  - From general principles to specific examples.
  - Example:

    Theory - All humans are mortal
    Hypothesis - Is Aman mortal?
    Observation - Aman is a human
    Confirmation - Aman is mortal



Concept: Algorithms

- What is an algorithm?
  - A sequence of steps for solving a problem.

- Why is it important?
  - In daily life, we use algorithms to describe solutions to problems.
  - Computer programs execute algorithms to perform specific tasks.

Algorithm for a PB&J sandwich

Materials:

- A jar of peanut butter
- A jar of jelly
- A loaf of sliced bread
- One butter knife

Your Task:

- What are the steps to make a peanut butter and jelly sandwich?

Concept: Debugging

- What is debugging?
  - Locating and fixing “bugs” in algorithms and processes to make them behave as expected.

Clicker Question 7

Scenario: You come home and the desk lamp in your apartment stopped working (it worked in the morning).
Clicker question 4: What is your first step to solve the problem?

A. Check if the lamp is turned on

B. Check if the light bulb is working

C. Check if the lamp is plugged in

D. Check if the outlet is working

E. Check if there is power in the room

Clicker Question 8

You checked A-E and it is still not working.
Clicker question 5: What do you do next?

A. Buy a new lamp (for example, -->)

B. Call your mother/friend/landlord/etc.

C. Use your roommate's lamp

D. Repeat steps A-E from before

E. Forget about the problem for the day

Summary: Questions

- Ask (yourself or the next person):
  - What is computational thinking (CT)?
  - What are the key elements of CT?
    - Or, what distinguishes CT from other types of problem solving strategies?
  - What makes CT important in K-12 education?

Summary: Answers

- What is CT?
  - **Alternative definition:** An approach to problem solving, which uses abstraction to create algorithmic solutions that can be automated with computational processes.
- Key elements:
  - Abstraction
  - Logical thinking
  - Algorithm
  - Debugging
- Importance in K-12:
  - You'll find out in the next lecture!

References

- J.M. Wing, “Computational Thinking,” CACM viewpoint, vol. 49 no. 3, March 2006, pp. 33-35.
- J.M. Wing, “Computational Thinking and Thinking About Computing,” Philosophical Transactions of the Royal Society, vol. 366, July 2008, pp. 3717-3725.
- P. Phillips, “Computational Thinking: A Problem-Solving Tool for Every Classroom,” in NECC 2007.
- Ormrod J. Educational Psychology: Developing Learners (7th Edition). New York, NY: Pearson, 2011.

- Computer Science K-12 Education
  - http://www.csta.acm.org/
  - http://csunplugged.org/
  - http://www.cs4fn.org/
  - http://www.csedweek.org/