# ACM SIGCSE 2007
## Special Session
# Computational Thinking

Jeannette Wing, Carnegie Mellon University

Peter Henderson, Butler University

Orit Hazzan, Technion – Israel Institute of Technology

Tom Cortina, Carnegie Mellon University

# Outline

- An Overview of Computational Thinking
  - Jeannette Wing
- Potential Influence of Computational Thinking on K-12 Education
  - Peter Henderson
- Potential Influence of Computational Thinking on Undergraduate Education
  - Orit Hazzan
- An Introductory Course Using Computational Thinking Principles
  - Tom Cortina
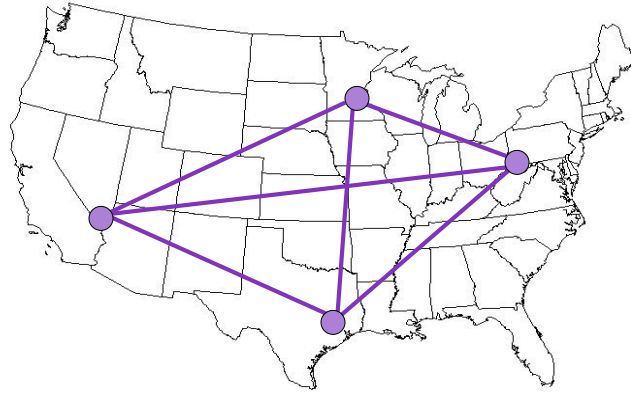- Audience Questions and Discussion

## An Undergraduate Course Using Computational Thinking Principles

- Principles of Computation

- Introduced in 2005 at Carnegie Mellon University for non-majors as a broader introduction to the field of computer science.

- Principles of Computation focuses on studying the process of computation, not necessarily on a computer.

- Does <u>not</u> include the traditional development of programming skills with a modern programming language.

## Examples from Principles of Computation

- Add up a set of n numbers in a list.
  - ITERATIVE solution
    1. Initially set the sum to 0.
    2. Initially set an index i to 1.
    3. If i ≤ n, do the following:
       a. Add the $i^{th}$ number of the list to the sum.
       b. Add 1 to i.
       c. Go back to step 3.
  - RECURSIVE solution
    - If there's only one number in the list, that's the sum.
    - Otherwise, add the first number in the list to the sum of the rest of the list.
- Which is easier to express? more intuitive?

**Examples from**
**Principles of Computation**



---

**Examples from**
**Principles of Computation**

- We need to fly to n cities and make sales presentations at each city. We need to visit each city only once and return to our home city. An airline offers direct flights between each pair of cities we wish to visit.
- What is the cheapest way to do this?
  - ◆ Seems simple enough: Find every possible route through the cities that starts and ends with the home city, and determine which one is cheapest.
  - ◆ Let students figure out the solution for 4 cities.
  - ◆ Then give them 40 cities.
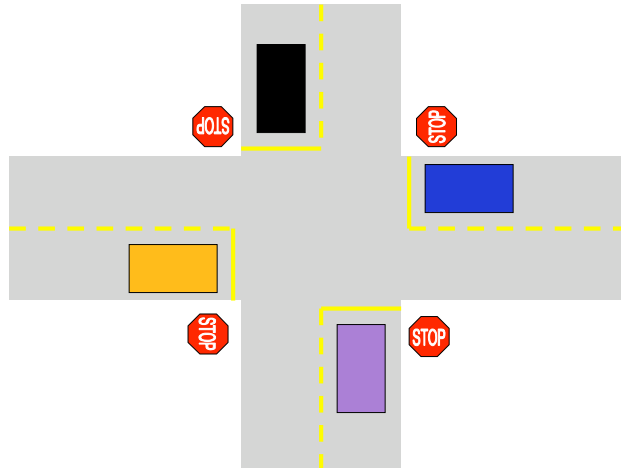- Can computers really solve everything for us?

## Examples from
## Principles of Computation



## Examples from
## Principles of Computation

- We need to wash, dry and iron 5 loads of laundry. We only have one washer, one dryer and one iron.
  It takes 90 minutes to wash, dry and iron one complete load of laundry.
- How can we get all of the laundry done in less than 450 minutes?
  - ◆ Use the notion of pipelining. After the first load of laundry has been washed and begins to dry in the dryer, start washing the second load of laundry.
  - ◆ When do we get the shortest total time to do the laundry using pipelining?
- How is pipelining used in computers?

## Examples from
## Principles of Computation



---

## Examples from
## Principles of Computation

- Four cars stop at an intersection with a four-way stop sign at the same time.
- Who should proceed first?
  - ◆ Laws state that the person on your right should proceed first, but what happens if everyone followed this law exactly?
  - ◆ We end up in a deadlock situation.
- How have computer scientists come up with ways to detect and correct situations where deadlock can occur?
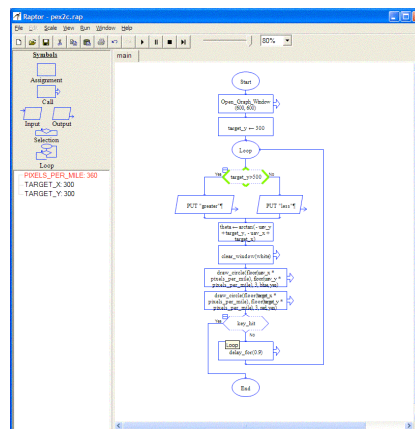- Why are computer scientists concerned with deadlock?

# Examples from
# Principles of Computation



- When we order something from an online merchant, how safe is our credit card number as it travels from our computer to the merchant's computer?
- How is our data encrypted so that it cannot be decrypted if someone else gets the information?
- How do public-key encryption systems work computationally?

# Programming without Syntax

- Our course uses a public-domain program called RAPTOR, a flowchart simulator.
- Students can build simple procedural programs without learning the syntax details of a language.
- Contains conditionals, loops, input and output, arrays, subroutines, graphics.



http://raptor.martincarlisle.com/

# For More Information

- **"An Introduction to Computer Science for Non-majors using Principles of Computation"**
- Friday, March 9
- 10:30-11:45AM
- Ballroom B

**SHAMELESS PLUG**