

Wrapping it all up

- Where have you been?
 - What are the key topics from this course?
 - What are the key lessons to take from this course?
- Where are you headed?
 - How might you use the knowledge you have gained?

What do computer scientists do?

- They think computationally
 - Abstractions, algorithms, automated execution
- Computational thinking will be a fundamental skill used by everyone in the world by the middle of the 21st Century
- Just like the three r's: reading, 'riting, and 'rithmetic
 - Ubiquitous computing and computers will enable the spread of computational thinking as a fundamental skill for every well educated person



Computational Thinking: the Process

- Identify or invent useful abstractions
 - Suppressing details, formulating interfaces
- Formulate solution to a problem as a computational experiment using abstractions
- Design and construct a sufficiently efficient implementation of experiment
- Validate experimental setup (i.e., debug it)
- Run experiment
- Evaluate results of experiment
- Repeat as needed

The three A' s of computational thinking

- Abstraction
 - Choosing the right abstractions
 - Operating in terms of multiple layers of abstraction simultaneously
 - Defining the relationships the between layers
- Automation
 - Think in terms of mechanizing our abstractions
 - Mechanization is possible
 - Because we have precise and exacting notations and models
 - There is some “machine” that can interpret our notations
- Algorithms
 - Language for describing automated processes
 - Also allows abstraction of details

Examples of computational thinking

- How difficult is this problem and how best can I solve it?
 - Theoretical computer science gives precise meaning to these and related questions and their answers
- Thinking recursively
 - Reformulating a seemingly difficult problem into one which we know how to solve
 - Reduction, embedding, transformation, simulation



