



CPSC 100

Computational Thinking

Application of CT

Instructor: Parsa Rajabi
Department of Computer Science
University of British Columbia



Agenda

- Intro to Computational Thinking (CT)
 - Continue class activity
 - Discussion
- Algorithms

Learning Goals



Learning Goals

After this lecture, you should be able to:

- Apply CT subskills to design and execute a structured solution (continued)
- Explain/define the concept of Algorithms
 - Describe its relevance to CT and where it originated from
- Define the concepts of *decomposition*, *abstraction* and *synthesis* in relation to an algorithm

Activity Review



Class Activity: Sort the Cards

Imagine a robot must arrange a set of cards in ascending order (Ace to King, Same suit).

The robot can only follow your instructions.

Task [Groups of 2-3]

Create a clear set of steps/instructions to sort the cards





Join Miro Board

<http://tiny.cc/CPSC-100-W1B>

Miro works best on laptop/iPad



Class Discussion



Algorithms

Algorithms

Muhammad ibn Musa al-Khwarizmi
(*Algorithmi*)

- Persian mathematician around 800 CE
- Discussed how to formulate mathematical procedures





Algorithms

An *algorithm* describes a sequence of steps that is:



Algorithms

An ***algorithm*** describes a sequence of steps that is:

1. Unambiguous

- No “assumptions” are required to execute the algorithm
- The algorithm uses precise instructions



Algorithms

An ***algorithm*** describes a sequence of steps that is:

1. Unambiguous

- No “assumptions” are required to execute the algorithm
- The algorithm uses precise instructions

2. Executable

- The algorithm can be carried out in practice



Algorithms

An ***algorithm*** describes a sequence of steps that is:

1. Unambiguous

- No “assumptions” are required to execute the algorithm
- The algorithm uses precise instructions

2. Executable

- The algorithm can be carried out in practice

3. Terminating

- The algorithm will eventually come to an end, or halt



To sort, you create an *algorithm*



To sort, you create an *algorithm*

An *algorithm* is a precise, systematic method for producing a specified result.



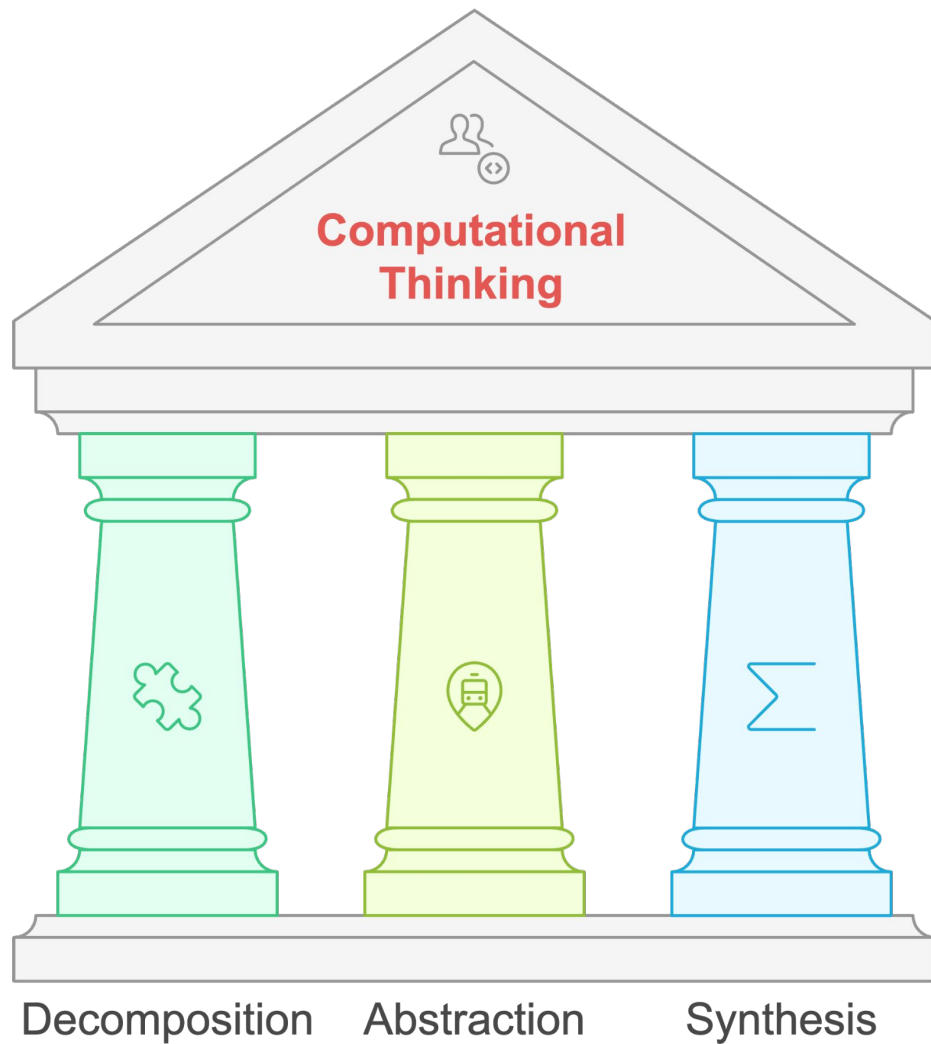
To sort, you create an *algorithm*

An *algorithm* is a precise, systematic method for producing a specified result.

Your sorting algorithms used:

- Decomposition*
- Abstraction*
- Synthesis*

**definitions in the appendix*



Decomposition

"If you can't solve a problem, then there is an easier problem you can solve: find it."

- George Polya

Abstractions



"The **most important** and high-level thought process in computational thinking is the **abstraction process**."

- Jeannette Wing

"The **most important** and high-level thought process in computational thinking is the **abstraction process**.

Abstraction is used in **defining patterns, generalizing from instances, and parameterization**. It is used to **let one object stand for many**."

- Jeannette Wing

Synthesis

"We are drowning in information, while starving for wisdom. The world henceforth will be run by **synthesizers**, people able to **put together the right information at the right time**, think critically about it, and make important choices wisely."

- Edward O. Wilson



iClicker

Set up your iClicker

- Set up an iClicker account.
 - You may already have done so
- Add this course to your account from Canvas.
- “Join” today’s class at student.iclicker.com



Set up your iClicker

www.join.iclicker.com/IFKA



Q: In the context of algorithms, _____ is a way of describing the solution in a general manner.

- A. Decomposition
- B. Synthesis
- C. Computational Thinking
- D. Abstraction
- E. Encapsulation



Wrap up

This is **NOT** a
"GPA booster class"



A+ is rare in this course



Wrap Up

- Labs will begin starting next week!
 - Go find ICCSx050 **before Wednesday**
- Join Ed Discussion board (link on canvas)
- Complete course survey via Canvas
 - Due January 13
- Go outside and enjoy the first weekend!

**Take Home
Slides**

To sort, you create an *algorithm*

An *algorithm* is a precise, systematic method for producing a specified result.

Your sorting algorithms used:

- Decomposition
- Abstraction
- Synthesis

To sort, you create an *algorithm*

An *algorithm* is a precise, systematic method for producing a specified result.

Your sorting algorithms used:

- **Decomposition:** breaking down the problem into smaller tasks you could solve

To sort, you create an *algorithm*

An *algorithm* is a precise, systematic method for producing a specified result.

Your sorting algorithms used:

- **Abstraction:** describing the solution in a general way that's applicable no matter what order the cards are in initially

To sort, you create an *algorithm*

An *algorithm* is a precise, systematic method for producing a specified result.

Your sorting algorithms used:

- **Synthesis:** putting all the steps together to solve the whole problem