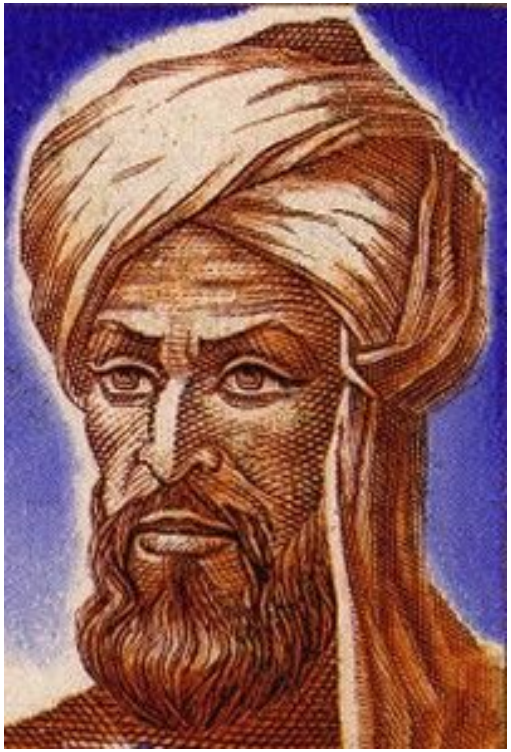
Slides for Pre-reading

Slides with yellow borders will not be covered in class, but are still testable content - you should review this before class.

Algorithms

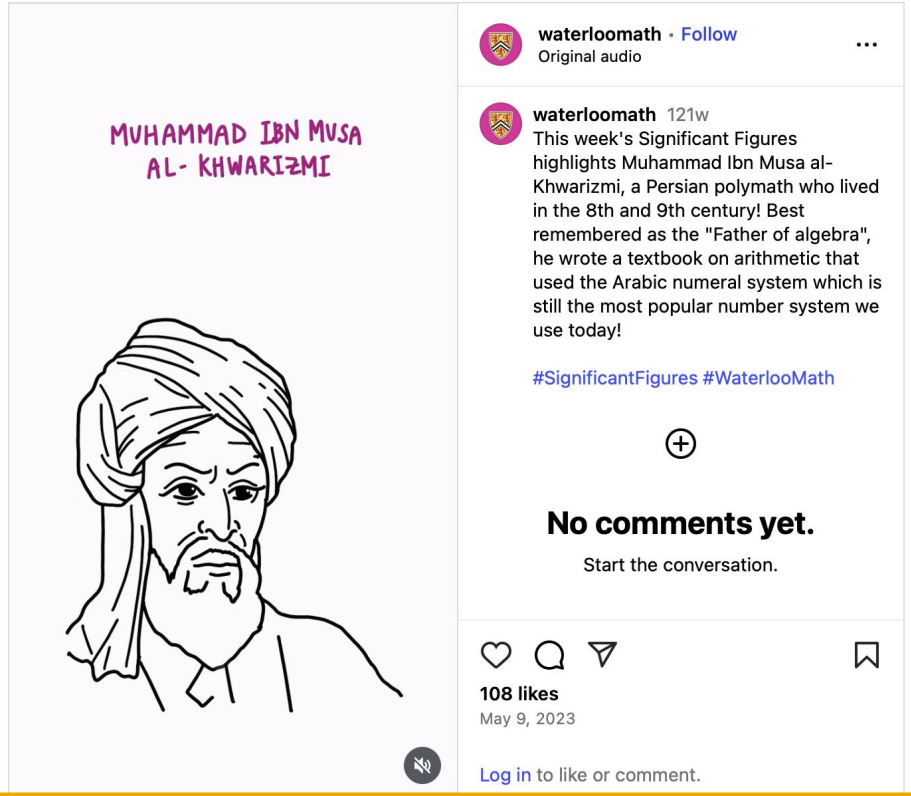
Muhammad ibn Musa al-Khwarizmi (*Algorithmi*)

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



Algorithms - Check out this Reel

Instagram



To sort, you create an *algorithm*

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

Your sorting algorithms used:

- Decomposition
- Abstraction
- Evaluation

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:

- **Evaluation**: Assessing a solution to a problem and using that information again on new problems.

Moore’s Law

Moore’s Law

Gordon Moore one of the co-founders of **Intel** corporation established the term "Moore's Law" in 1965, This law explains how the number of transistors on integrated circuits is increasing exponentially, which boosts computing capability and lowers prices. Moore's law has had a significant and far-reaching impact on technology, from the introduction of personal computers and smartphones to the advancement of artificial intelligence and Internet of Things(IoT) devices. it has accelerated development in sectors including telecommunications, healthcare, transportation, enabling organizations and people to accomplish things that were previously unthinkable.

Moore's Law Definition

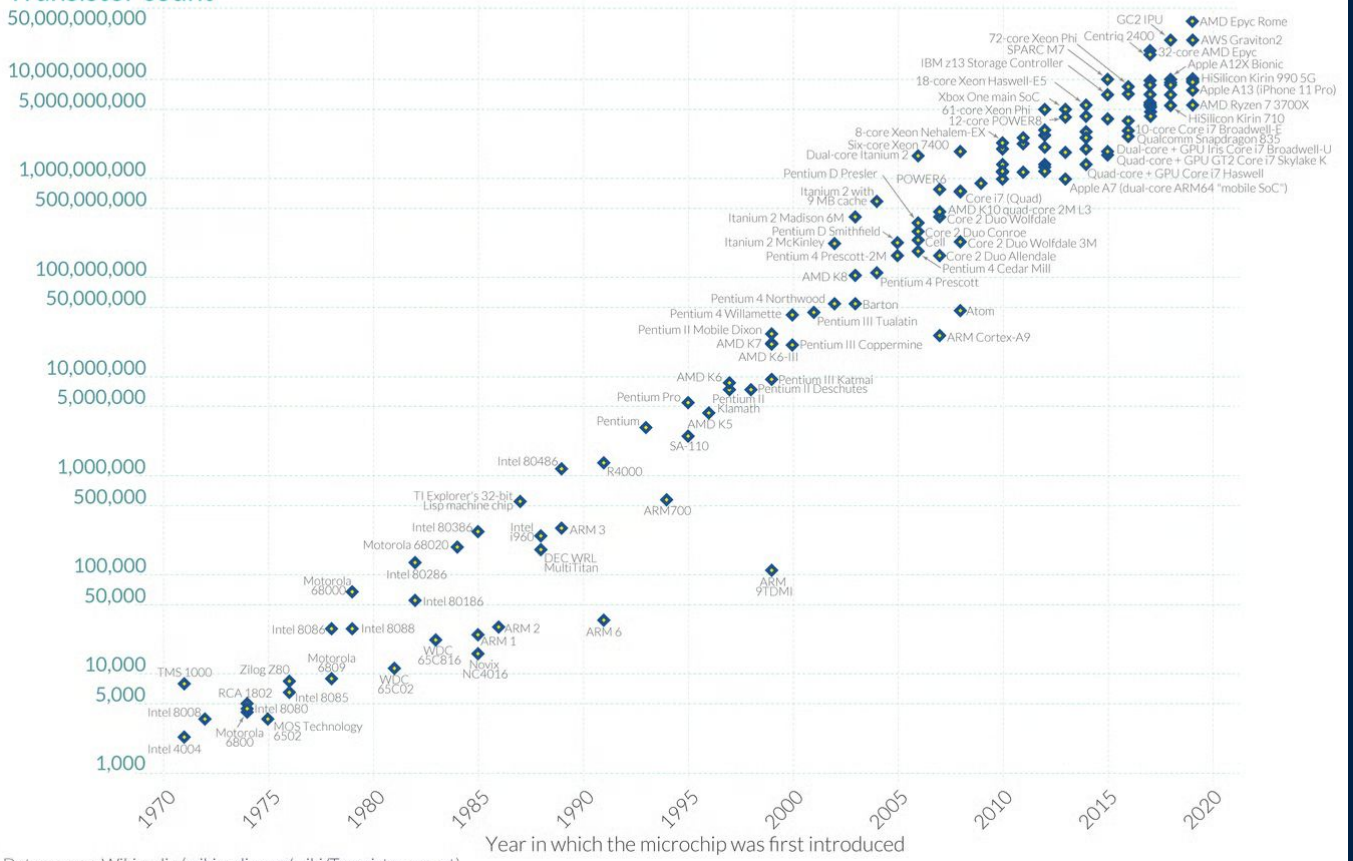
The exponential increase in the number of transistors on integrated circuits over time is referred to as Moore's law. According to this, a chip transistor count tends to double every two years or so, resulting in higher processing power and better performance.

Source

Moore's Law: The number of transistors on microchips has doubled every two years

Our World in Data

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years.
This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.



Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)
OurWorldinData.org – Research and data to make progress against the world's largest problems.

CPSC 100

Computational Thinking

Application of CT: Algorithms

Instructor: Dr. Firas Moosvi
Department of Computer Science
University of British Columbia

Course Admin

Agenda

- Applying Computational Thinking (CT)
  - Sorting Class activity
  - Discussion
- Introduction to Algorithms

Learning Goals

Learning Goals

After this lecture, you should be able to:

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

Class Activity

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 3-4]

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

Class Discussion

Participation Question

Did you come up with a list of steps to sort cards?

Successful: What worked?	Unsuccessful: Where did it fail ?
Logistics	Counter Example

Algorithms

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*

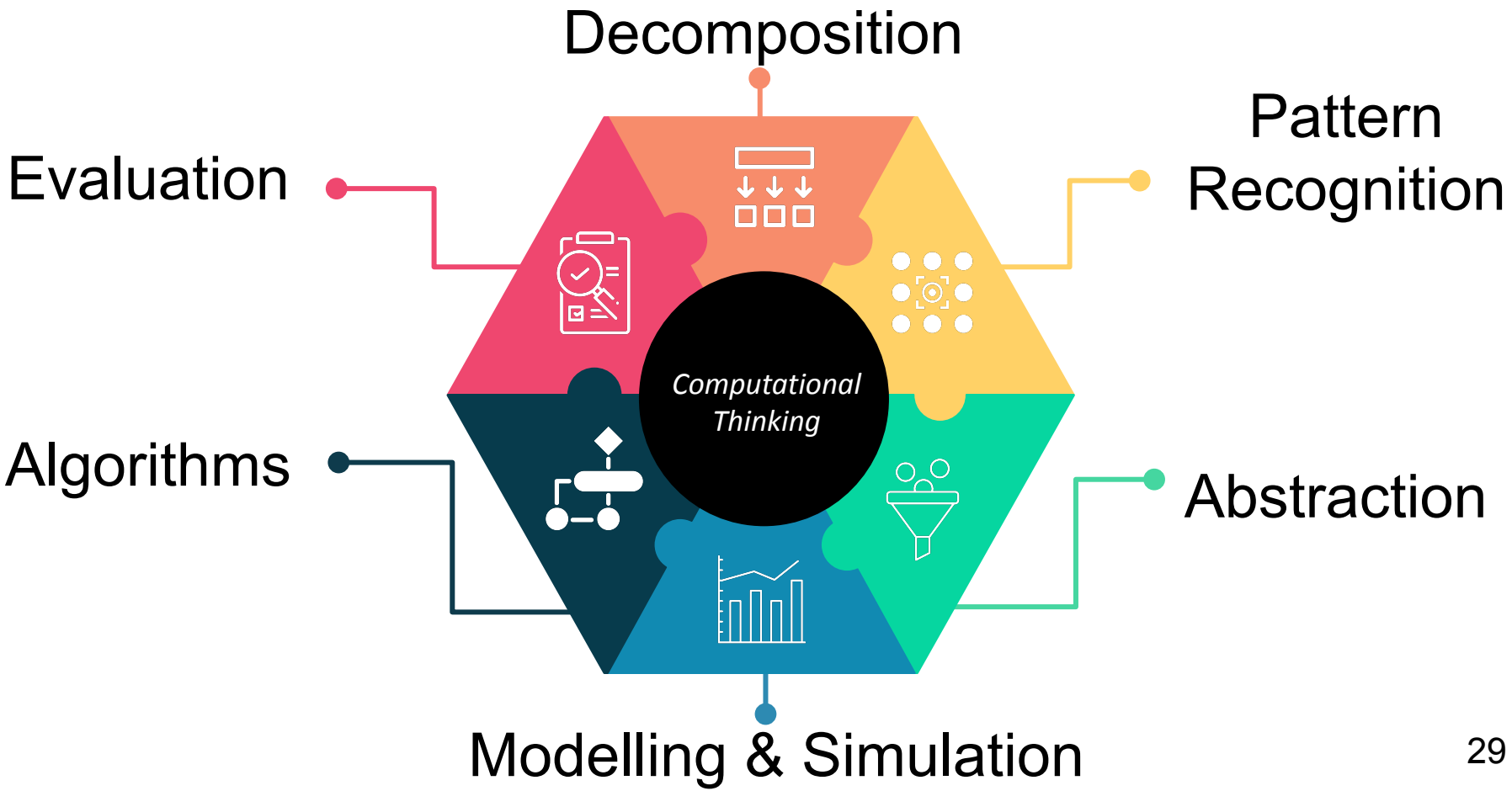
An ***algorithm*** is a precise, systematic method for producing a specified result.

Your sorting algorithms used:

- Decomposition*
- Abstraction*
- Evaluation*

**definitions in the appendix*

Computational Thinking Skills



DECOMPOSITION

Breaking down problems into smaller, easier parts.

PATTERN RECOGNITION

Using patterns in information to solve problems.

ABSTRACTION

Finding information that is useful and taking away any information that is unhelpful.

MODELLING AND SIMULATION

Trying out different solutions or tracing the path of information to solve problems.

ALGORITHMS

Creating a set of instructions for solving a problem or completing a task.

EVALUATION

Assessing a solution to a problem and using that information again on new problems.

Decomposition

DECOMPOSITION

Breaking down problems into smaller, easier parts.

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

- George Polya

Abstraction

ABSTRACTION

Finding information that is useful and taking away any information that is unhelpful.

"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

Evaluation

EVALUATION

Assessing a solution to a problem and using that information again on new problems.

"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."

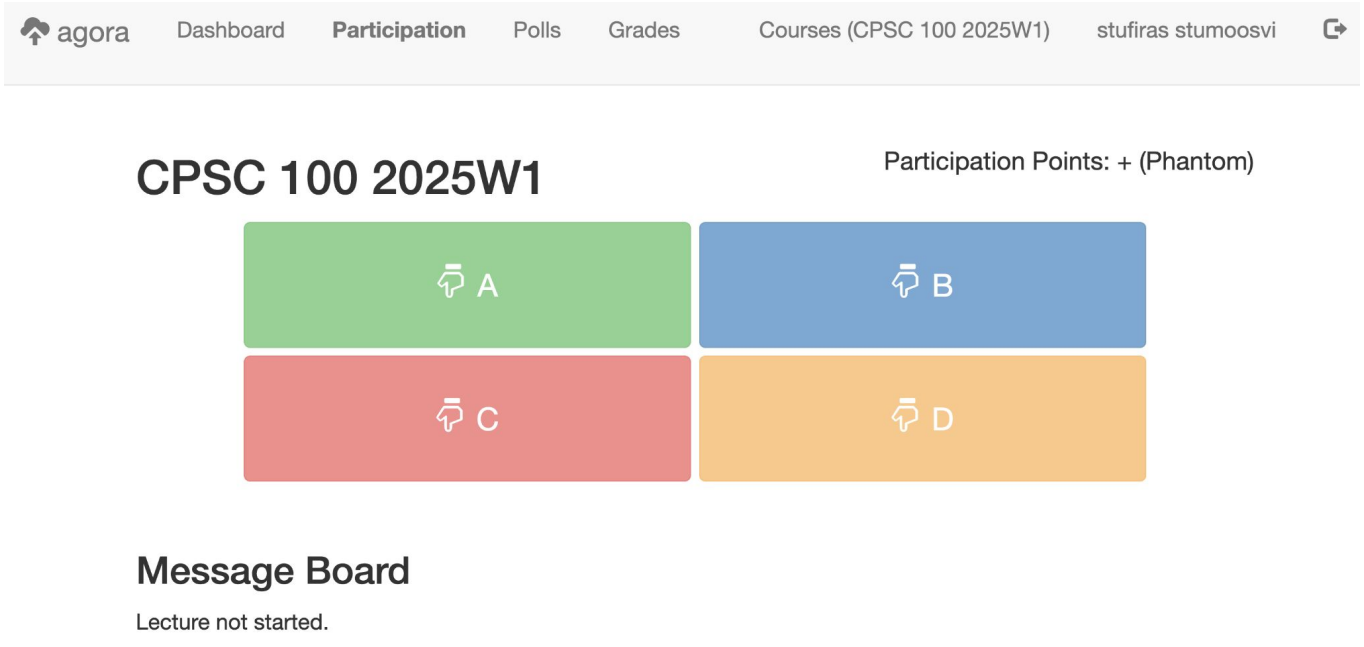
- Edward O. Wilson

Clicker Questions (using Agora)

Agora: Discussions and Polling

Page: agora.students.cs.ubc.ca (login with your CWL)

Enroll code: **psychohistory**



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

A. Decomposition

B. Evaluation

C. Computational Thinking

D. Abstraction

E. Encapsulation

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

A. Decomposition

B. Evaluation

C. Computational Thinking

D. Abstraction

E. Encapsulation

Wrap up