

Introduction to Computational Thinking

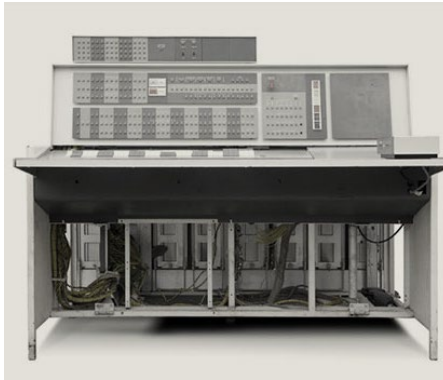
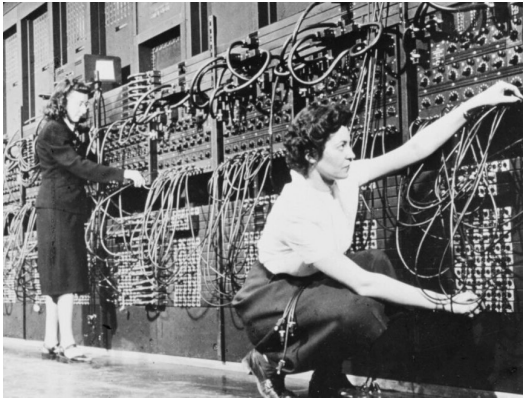
Alexander Lam

What is Computing?

Computing is any goal-oriented activity requiring, benefiting from, or creating computing machinery (**computers**). (Wikipedia)

What is a Computer?

A **computer** is a machine that can be programmed to automatically carry out sequences of arithmetic or logical operations (**computation**). (Wikipedia)



What is a Computer?

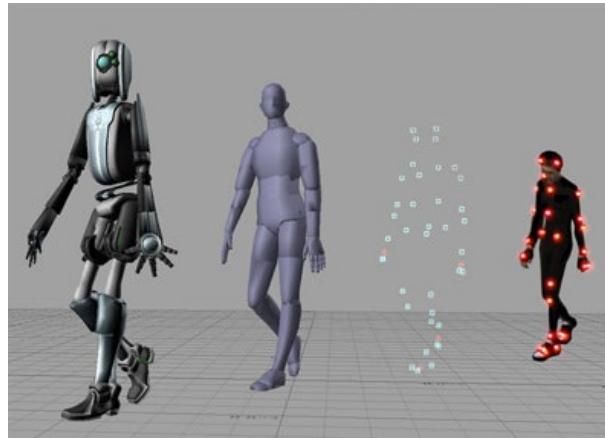
A **computer** is a machine that can be programmed to automatically carry out sequences of arithmetic or logical operations (**computation**). (Wikipedia)

- Computers are good at doing what they have been programmed to do
 - Smart
- Computers are bad at doing what they have **not** been programmed to do
 - Dumb

To be good at programming, it is important to understand this distinction

The Limitations of Computers

- Computer programs have restricted inputs



The Limitations of Computers

- Computer programs have restricted outputs
 - Audio?
 - Visual?
 - Textual?



GPT-4.1 (2025-04-14)

I can't directly create and send files, but I can provide you with instructions and the content you need to easily generate an Excel file with numbers 1 to 10.

The Limitations of Computers

- Computer programs have restricted operations
 - System memory size
 - Processing limitations
 - Skill of the programmer

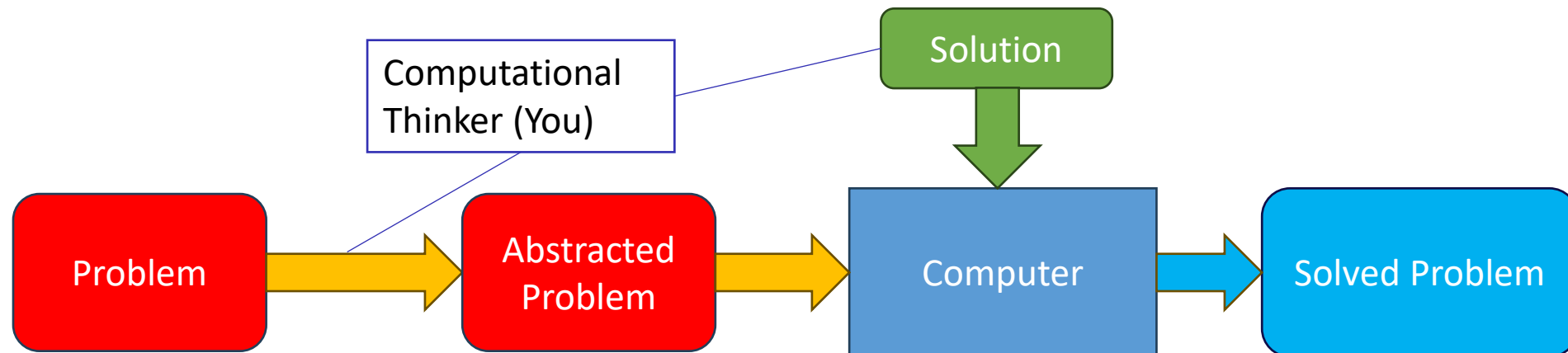
Benefits of Computers

- Computers are:
 - Fast
 - Accurate
 - Repetitive
 - Structured

Computational Thinking

Computational Thinking is an approach that formulates **solutions** to **problems**, and expresses them in a way such that a **computer** can execute them.

- **Problem** needs to be in a format interpretable by the computer.
- **Solutions** are in the form of computational operations.



Computational Thinking

Computational Thinking is an approach that formulates **solutions** to **problems**, and expresses them in a way such that a **computer** can execute them.

- It is **not** just about **writing programs**.
- It is **not** about **thinking like a computer**, but rather making the problem manageable to a computer.
 - Knowing how to think like a computer can help though
 - Empathize with the computer

Computational Thinking (alt. definition)

Computational thinking is the process of approaching a **problem** in a systematic manner, and creating and expressing a **solution** such that it can be carried out by a **computer**.

Computational Thinking

Viewpoint | Jeannette M. Wing

Computational Thinking

It represents a universally applicable attitude and skill set everyone, not just computer scientists, would be eager to learn and use.



Computational thinking builds on the power and limits of computing processes, whether they are executed by a human or by a machine. Computational methods and models give us the courage to solve prob-

cisely. Stating the difficulty of a problem accounts for the underlying power of the machine—the computing device that will run the solution. We must consider the machine’s instruction set, its resource constraints, and its operating environment.

In solving a problem efficiently, we might further ask whether an approximate solution is good enough, whether we can use randomization to our

Computational Thinking

- **Abstraction:** Problem formulation
 - Understand and represent the problem
- **Automation:** Solution expression (**programming**)
 - Design the program steps
- **Analysis:** Solution execution and evaluation
 - Develop and run program

A Scenario

You're a manager at a software company, and at 6 pm on a Friday, the intern working on a problem tells you that they've:

- **Abstracted** the problem such that a computer program can interpret it.
- Written a program to solve the problem **automatically** and **correctly**.
- **Analysed** the program to ensure that it **always** solves the problem **correctly**.

Is this good enough for you?

The Problem in Question

Problem: Sort every employee in order of their birthdays.



e.g. if John's birthday is April 5 and Alice's birthday is May 17, John should be before Alice

Intern's Abstraction: Program takes as input the employees' names and their associated birthdays.

Intern's Program:

- Rearrange the employees in a random order.
- Check if the employees are correctly ordered by their birthdays.
- If yes, we are done.
- If no, repeat the program.

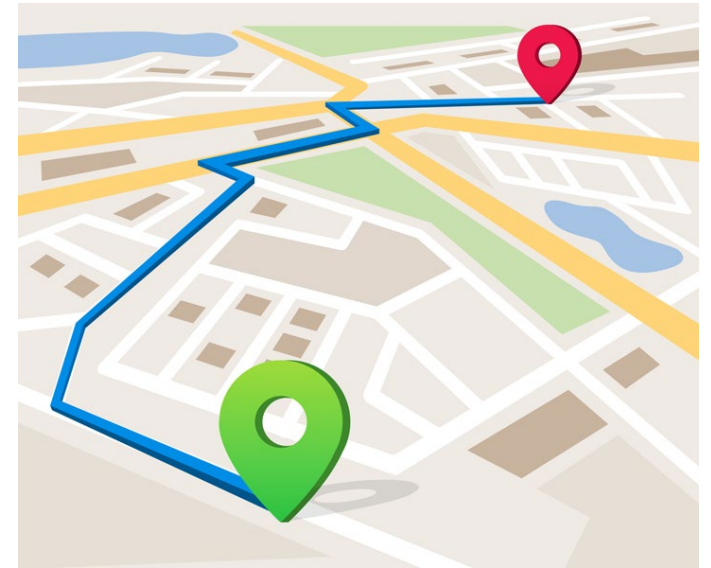
Time Efficiency

It's not enough for the program to correctly solve the problem.

The program also needs to solve the problem in a reasonable amount of time.

Examples of Problems

- Multiplying two numbers
- Finding the optimal route
- Predicting the Stock Market
- Verifying that you are a real person



Why Computer Science?

Broad spectrum of jobs

- Engineering
- **Finance – Fintech**
 - eCommerce
 - Quant Firms
 - Banks
 - Data Science
- Artificial Intelligence
- Enterprise Software Development
- Cybersecurity
- **“Big Tech”**

Why Computer Science?

“Fun”

- Some problems can be “nice” to think about
- Satisfaction of finding solutions to difficult problems
- Forefront of modern technology
 - AI Revolution
- Different perspective of the world
 - Represent the world in a logical, structured manner

