Lecture 01 — Introduction to Algorithms

COMPSCI 308 — Design and Analysis of Algorithms

Xing Shi Cai https://newptcai.gitlab.io

Duke Kunshan University, Kunshan, China

Oct-Nov 2024

Summary

Assignments[1]

Practice makes perfect!

Introduction to Algorithms (ITA)

📖 Read —

- Section 1.1, 1.2

✏️ Practice —

- Exercises 1.1: 1–5
- Exercises 1.2: 1–3
- Problem 1-1
- Exercises in this slide deck

[1] 💣 Assignments will not be collected; however, quiz problems will be selected from them. (This includes both Readings and Exercises.)

What Is an Algorithm?

What Is an Algorithm?

An algorithm is any well-defined computational procedure that takes some values as input and produces some values as output.

🍰 What is the difference between an algorithm and a mathematical function?

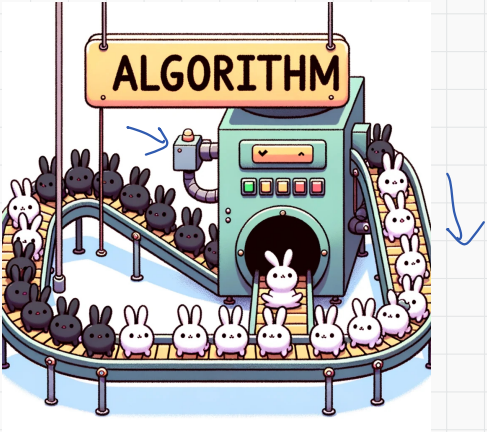


Figure 1: 😏 An algorithm that takes a black 🐰 as input and a white 🐰 as output.

📌 Example: Age Calculation

🍰 How do we design an algorithm with

- Input: birth date (y_1, m_1, d_1) and current date (y_2, m_2, d_2)
- Output: age

year month day

For example. if $y_1 = 2006$, $y_2 = 2026$

then the age is. $y_2 - y_1 = 2026 - 2006 = 20$

This is wrong. For example if $m_1 = 6$, $m_2 = 1$

The age is $y_2 - y_1 - 1 = 19$.

The correct algorithm.

If $m_1 < m_2$.

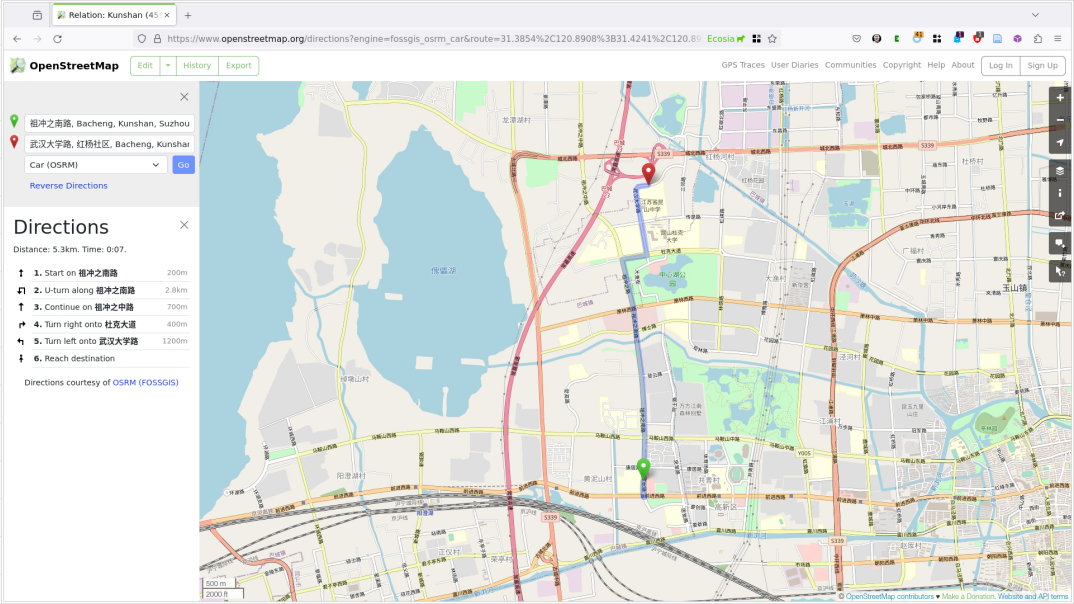
Return $y_2 - y_1$

If $m_1 = m_2$ and $d_1 \leq d_2$.

Return $y_2 - y_1$

Return $y_2 - y_1, -1$.

Example: Routing



📌 Example: 😷 Diagnosis

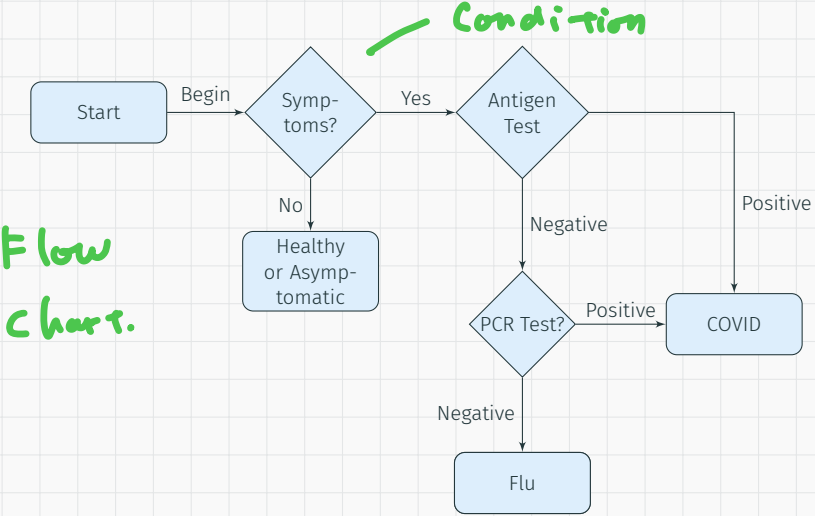


Figure 3: An algorithm to diagnose COVID

The Origin of Algorithms

No in exam.

Al-Khwarizmi was a polymath from Central Asia, who produced influential works in ➕, 🔭, and 🗺️.

Around 820 CE, he was appointed as the astronomer and head of the library of the *House of Wisdom* in Baghdad.

The word *algorithm* is derived from his name.

Figure 4: Al-Khwarizmi (780-850 CE)

Public Key Cryptography

😀 Nine Algorithms

The 👆 book *Nine Algorithms that Changed the Future (2012)* lists some algorithms which changed human history:

1. Search engine indexing
2. PageRank
3. Public-key cryptography
4. Forward error correction

😜 Nine Algorithms

The 👆 book *Nine Algorithms that Changed the Future (2012)* lists some algorithms which changed human history:

1. Search engine indexing
2. PageRank
3. Public-key cryptography
4. Forward error correction
5. Pattern recognition
6. Data compression
7. Database
8. Digital signature

🤔 Why are there only eight algorithms listed?

9. The algorithm that has not been invented yet.

The Problem with Unencrypted Messages

If you write a message to someone on a postcard, anyone who comes in contact with it can read the message.

The same holds true for any plain-text message sent through the internet.

Figure 5: Plain-text messages are not secure

What Is Encryption?

One way to secure your message is to encrypt the message.

Encryption is the process of converting a message or information into a coded form.

The transformation is done using an encryption key.

Figure 6: Encrypted messages are secure

📌 Example of an Encryption Algorithm

In the Caesar cipher, each letter in the plaintext is shifted a certain number of places down the alphabet cyclically.

For instance, with a shift of 2:

$$A \longrightarrow C, B \longrightarrow D, \ldots, Z \longrightarrow B,$$

The *key* is the position each letter is shifted.

🍰 What is the original message of the following text if the key is 2?

VJG RTQHGUUQT KU COCBKPI

📌 Example of an Encryption Algorithm

In the Caesar cipher, each letter in the plaintext is shifted a certain number of places down the alphabet cyclically.

For instance, with a shift of 2:

$$A \to C, B \to D, \dots, Z \to B,$$

The *key* is the position each letter is shifted.

🍰 What is the original message of the following text if the key is 2?

VJG RTQHGUUQT KU COCBKPI

Answer:

T → V , H → J, E → G

THE PROFESSOR IS AMAZING

The Problem with Symmetric Encryption

Using a previously exchanged 🔑 between the sender and receiver to encrypt messages is called symmetric encryption.

The challenge with symmetric encryption is to secure this key exchange.

Figure 7: Symmetric encryption requires exchange of keys

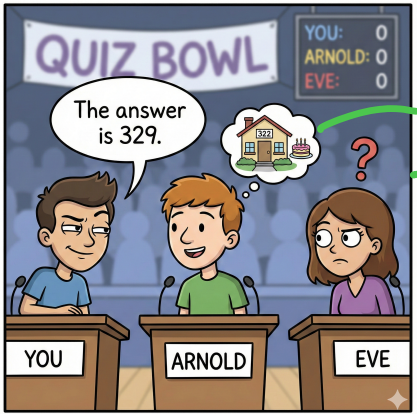
Example of Symmetric Encryption

In a quiz bowl, you want to tell your friend Arnold that the answer to a problem is 7.

But you don't want Eve, who is also there, to know the answer.

You and Arnold used to play in front of house number 322.

To use 322 as the key to send an encrypted message to Arnold— The answer is 329 minus the # of the house where we play.



329
−322
= 7

Figure 8: Eve cannot know the answer!

How to Exchange the Key Securely?

If the sender and receiver are strangers, the sender can encrypt the 🔑 itself and send it to the receiver.

But how can the receiver decrypt the key *without* any previous exchange of keys?

Figure 9: The receiver cannot decrypt the key without a shared key

How to Exchange the Key Securely?

If the sender and receiver are strangers, the sender can encrypt the 🔑 itself and send it to the receiver.

But how can the receiver decrypt the key *without* any previous exchange of keys?

The 🪄 of public key cryptography can make this happen.

Figure 9: The receiver can decrypt the key with public key cryptography

The Paint-Mixing Game

The paint-mixing game is a classic example of public key cryptography.

The aim of the game is for you and Arnold to each produce the same mixture of paint, without telling Eve how.

The 💡 is in this video.

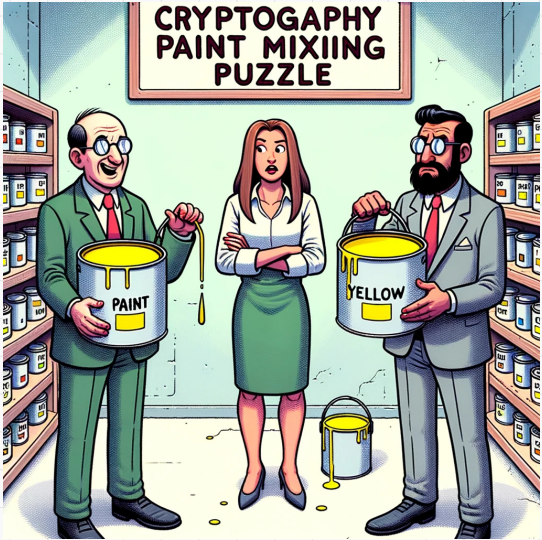
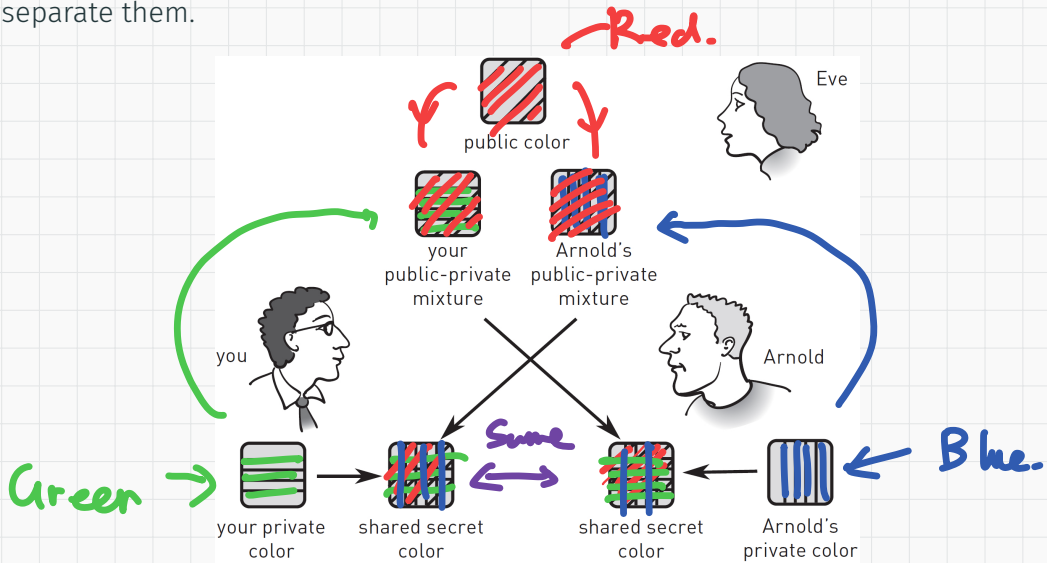


Figure 10: The paint-mixing game

The Trick of Paint Mixing

💡 What makes this work is that it is easy to mix two colours, but very difficult to separate them.



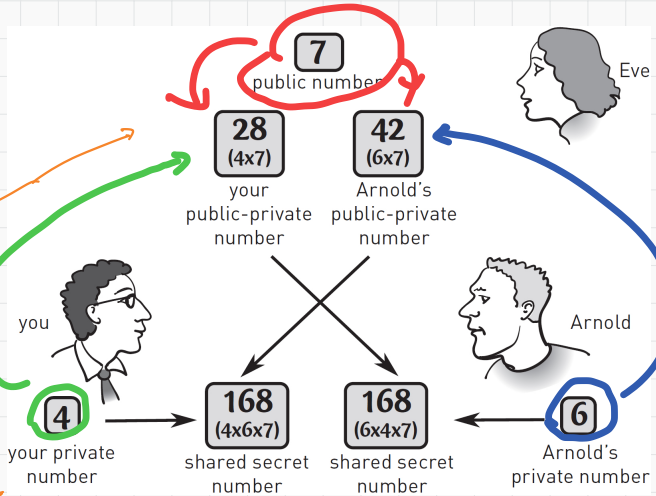
Translate Paints to Numbers

🤔 Using numbers instead of paints works in a similar way. Right?

This does not actually work.

Multiplication is easy to reverse.

In practice need to make this difficult to reverse.



Search Engine Indexing

Search Engine Indexing

Search engine indexing involves collecting, parsing, and storing web pages into a database for quick information retrieval.

Once indexing is done, a full-text search can quickly find relevant web pages.

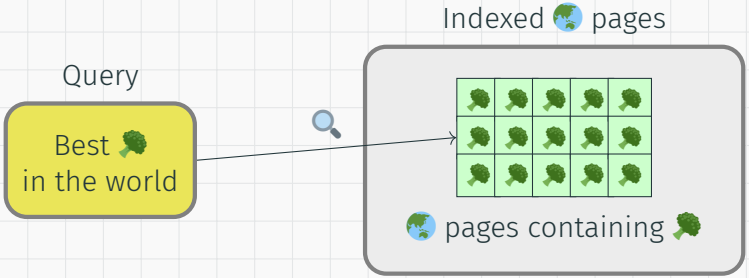


Figure 11: 🙂 All popular databases, such as `sqlite`, provide such a feature.

What Is PageRank? (Or — Why Google Exists?)

PageRank was developed by Larry Page and Sergey Brin.

It became a foundational algorithm for Google's search engine.

- PageRank aims to measure the importance of web pages.
- It works by counting both the number and quality of links to a page.

😩 Unfortunately, Search Engine Optimization (SEO) and Large Language Models (LLMs) have more or less made it obsolete.

Figure 12: Why is PageRank called PageRank?

Algorithm Design and Analysis

The Challenge of Algorithm Design

Create.

Designing the *right* algorithm can be challenging.

The space of choices in algorithm design is enormous.

Premature commitment to design choices often leads to loss of performance and limited flexibility.

Figure 13: There are too many ways to put things together!

To make an optimal algorithm, you should —

- Develop strong algorithmic skills.
- Gain expertise in the problem domain.
- Dedicate ample time for implementation.
- Engage in thorough experimentation.

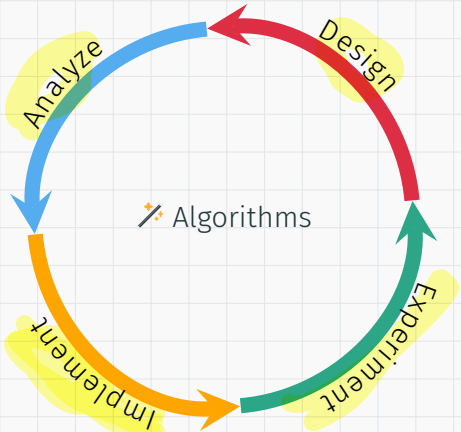


Figure 14: The cycle of algorithm design

The Analytical Engine, first described in 1837 by Charles Babbage, was proposed as a mechanical general-purpose computer.

By what course of calculation can these results be arrived at by the machine in the shortest time?

— C. Babbage (1864)

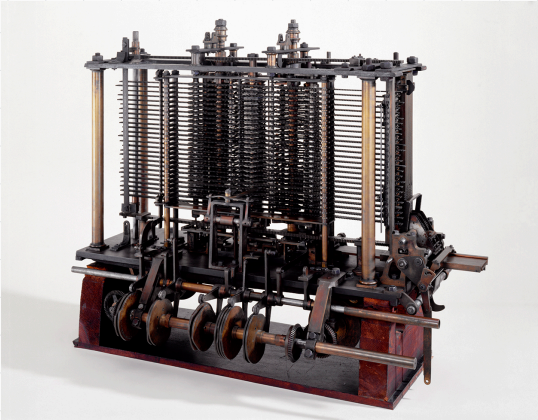


Figure 15: The Analytical Engine

The Modern Version of Algorithm Analysis

Common questions include —

- ⌚ How long will my program take?
- 😩 Why does my program run out of memory?

These questions are too vague and depend on many factors —

- 💻 The computer being used
- 1️⃣ The data being processed
- 🪢 The programming language that is doing the job

Figure 16: Why fast does not always mean efficient

Mathematical Rigour in Algorithm Analysis

Starting in the 1960s, people such as Donald Knuth, began rigorously analyzing algorithms through mathematical methods:

- ♾️ Formulate abstract models of computation.
- ✏️ Develop algorithms based on these models.
- 🔍 Analyze the time and space complexity using mathematical notation.
- ✅ Prove the correctness and optimality of algorithms.
- 🤔 Evaluate the theoretical limits.

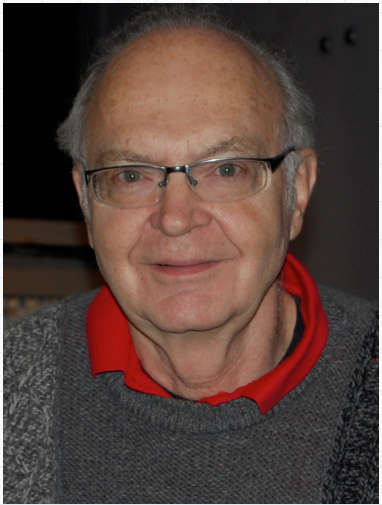


Figure 17: Donald Knuth (1938–) 🇺🇸💻👨‍🏫

The Art of Computer Programming (TAOCAP) is a comprehensive series of books written by Donald Knuth on algorithm analysis.

😲 The project started in 1962 and is still ongoing.

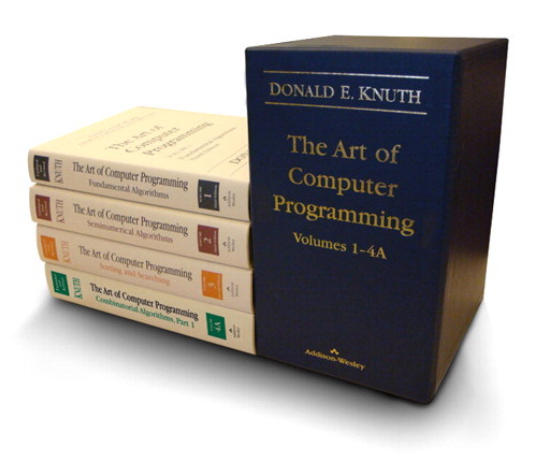
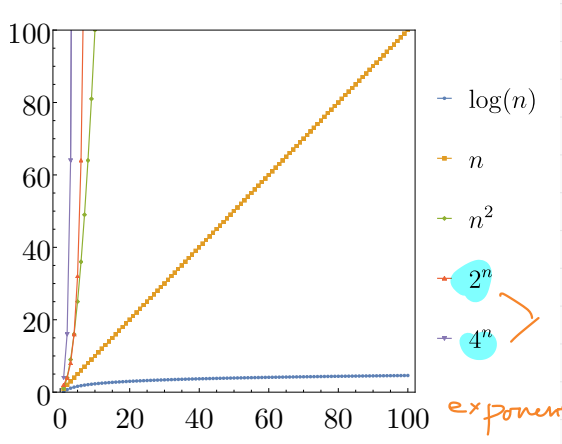


Figure 18: The Art of Computer Programming

A typical result of algorithm analysis is like this—

Given an input of size n, the algorithm has a running time $\log n$, n, or 2^n.

😲 The difference in efficiency between a $\log n$, n, or 2^n algorithm is huge.



The Fruit of Running Time Analysis

Algorithm analysis of running time serves as a cornerstone for understanding computational efficiency:

- 🔍 Provides insights into scalability.
- 🧐 Distinguishes between feasible and infeasible solutions.
- 💬 Offers a framework for discussing program behaviour.
- ✋ Extends performance lessons to other computational resources, such as memory usage.

🤔 Can you think of an application in which the speed of an algorithm is absolutely critical?

- High frequency Fading.
- Flight control.
- LLM.

What are your main takeaways today? Any questions?