INTRODUCTION TO COMPUTATIONAL THINKING

Understanding Concepts & Data Encoding and Representation

Compiled by Dr. DEJEY

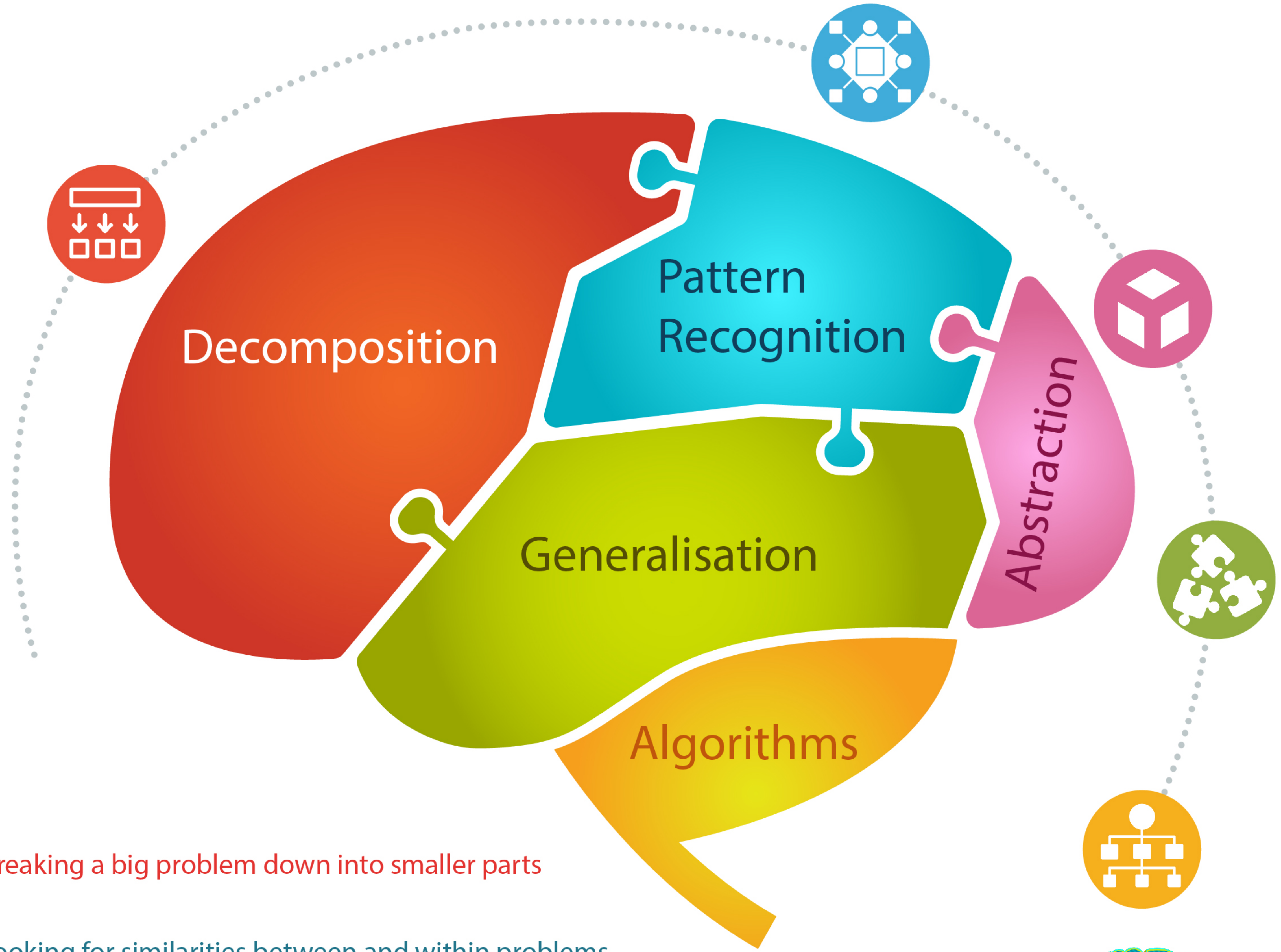
Overview of computational thinking and its importance

- Computational thinking is a **problem-solving process** that involves breaking down complex problems into manageable parts, recognizing patterns, developing step-by-step solutions, and generalizing these solutions for broader applications.
- It's not just about programming; it's **about a way of thinking that can be applied to various disciplines,** including science, mathematics, engineering, and everyday life.

Key Components of Computational Thinking

Computational Thinking

1. **Decomposition**: Breaking down complex problems into smaller, more manageable parts.
2. **Pattern Recognition**: Identifying similarities or patterns among and within problems.
3. **Abstraction**: Focusing on the important information only, and ignoring irrelevant detail.
4. **Algorithms**: Developing a step-by-step solution to the problem or the rules to follow to solve the problem.



- Breaking a big problem down into smaller parts
- Looking for similarities between and within problems
- Taking the detail out of a problem and ignoring irrelevant information
- Adapting solutions to other problems to solve new ones
- Simple rules to follow that solve the problem

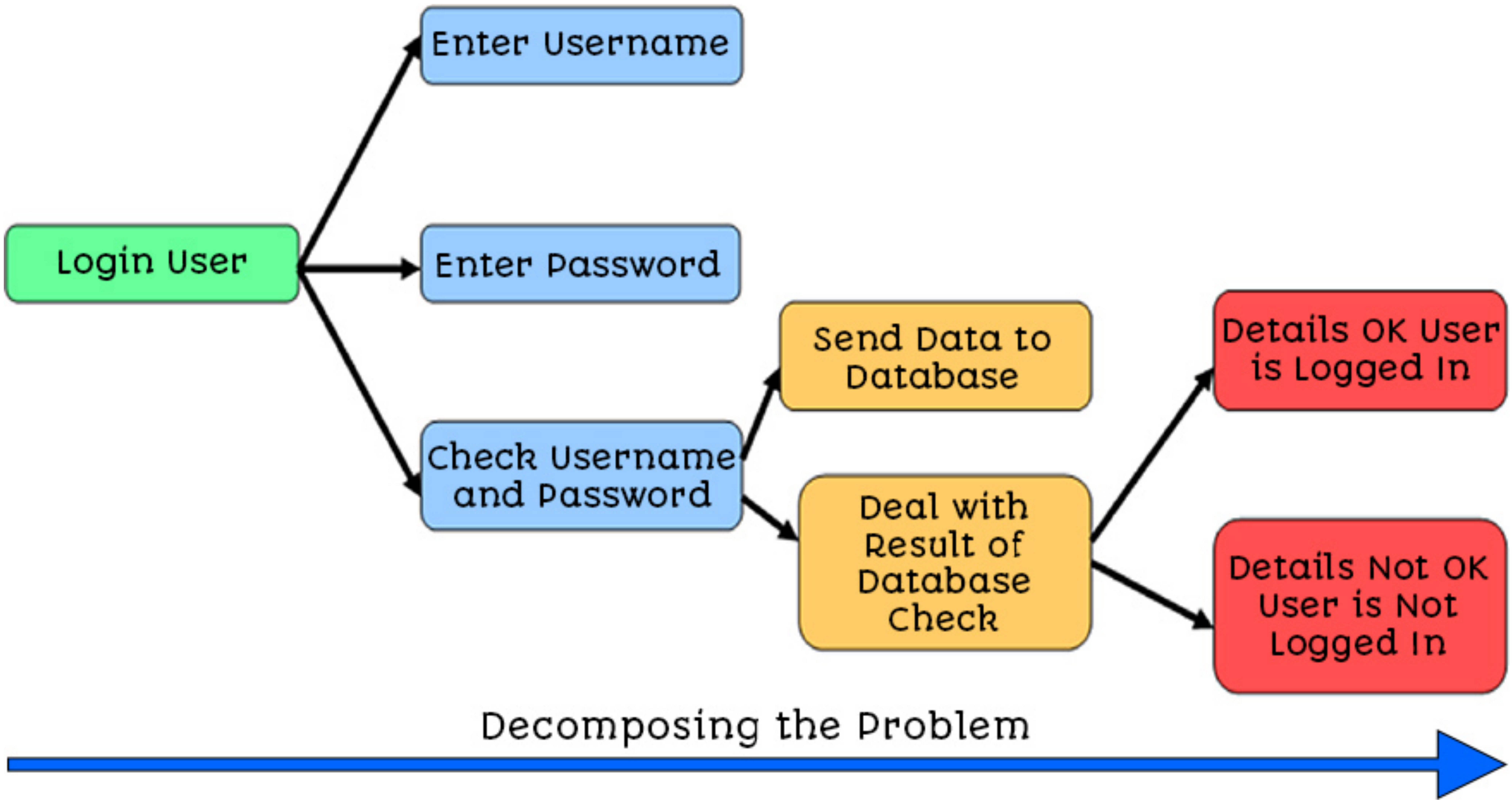
Decomposition

Decomposition involves breaking down a complex problem or system into smaller, more manageable parts. This makes the problem easier to understand and solve.

Application in Problem-Solving:

- **Example:** When **developing a new software application**, decomposition helps by breaking the project into smaller tasks such as UI design, database setup, backend logic, and testing. Each part can be handled separately, making the development process more organized and manageable.
- **Real-World Example: Planning a community event.** You can decompose the event into tasks like finding a venue, organizing food, inviting speakers, and arranging entertainment. By assigning these smaller tasks to different team members, the overall process becomes more efficient.
- **Example:** In **data analysis**, decomposition is used to **break down data sets** into smaller subsets. For instance, **analyzing sales data** can be divided by region, time period, product category, etc. This makes it easier to identify trends and patterns specific to each subset.

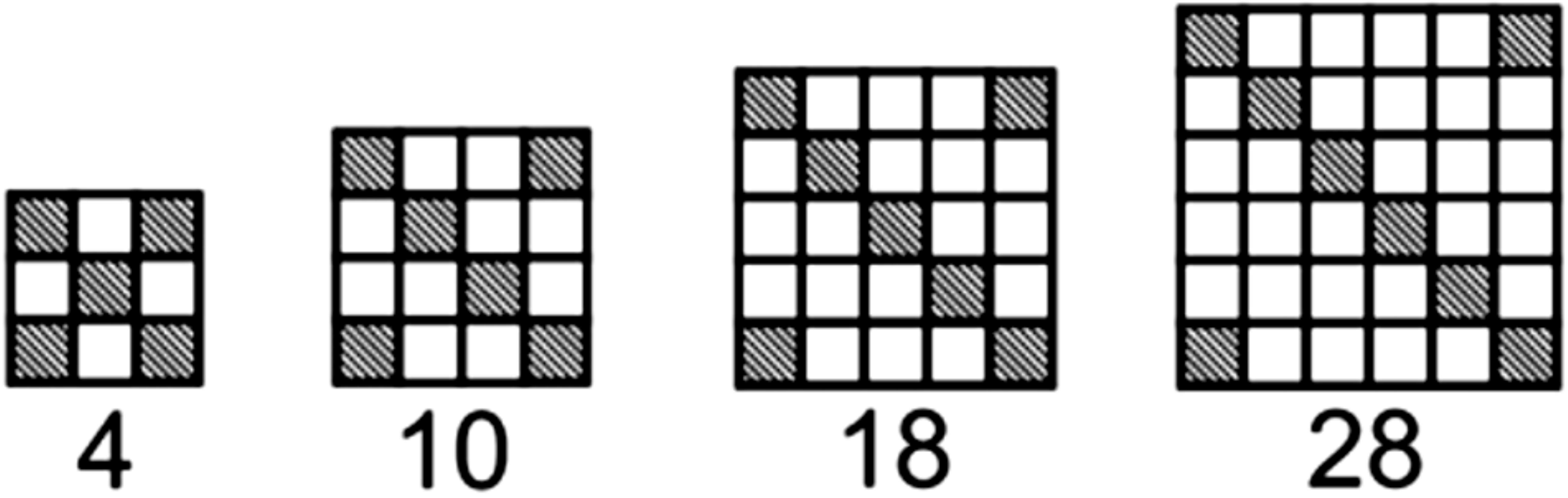
Decomposition



Pattern Recognition

- Pattern recognition involves **identifying similarities, trends**, or regularities in data. Recognizing patterns helps **predict future outcomes** and can simplify complex problems.
- **Application in Problem-Solving:**
  - **Example:** In **troubleshooting technical issues**, pattern recognition helps identify common symptoms that might indicate the same underlying problem, such as noticing that a software application crashes under similar conditions.
  - **Real-World Example:** In healthcare, **doctors** use pattern recognition to **diagnose diseases** by recognizing patterns of symptoms that are characteristic of certain illnesses.
  - **Example:** In **machine learning**, pattern recognition is essential for training algorithms to identify objects, recognize speech, or predict trends. For example, an e-commerce platform might recognize patterns in purchasing behavior to suggest products to customers.
  - **Example:** **Financial analysts** use pattern recognition to detect trends in stock market data, helping them make informed investment decisions.

Pattern Recognition

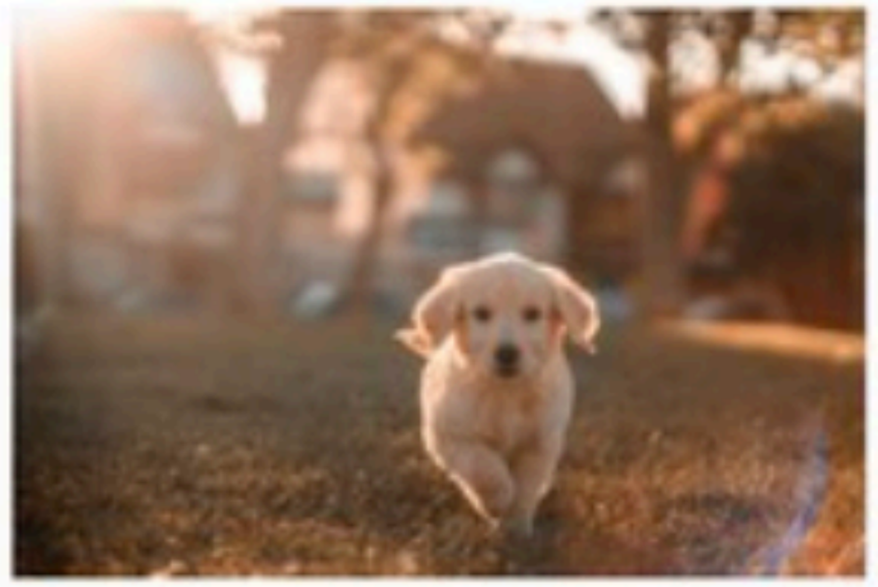


What number comes next?

Pattern Recognition

Train a model to distinguish whether a photo is about Cat or Dog

Cat Picture



Dog Picture

Similarity	Dissimilarity
4 Legs	Skin pattern
2 eyes	Food
Has a tail	Behavior
Has a neck	**Work in Police Station**
Has no wings	Family/Ordo

Trial 1 Training Feature
5 features: 70% Accuracy

Trial 2 Training Feature
10 features: 88% Accuracy

Trial 3 Training Validated Feature
8 features: 94% Accuracy

Training More Validated Features

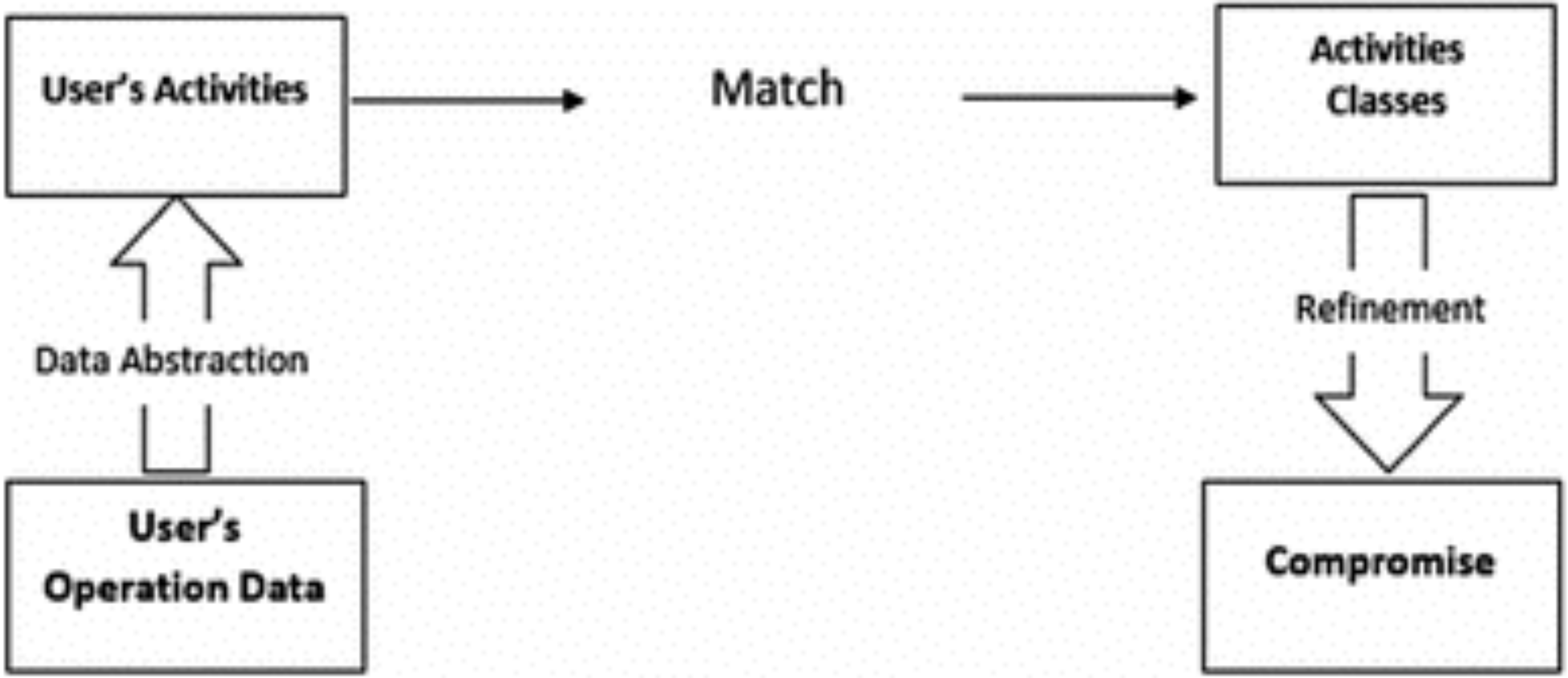
↓

Higher Model Accuracy

Abstraction

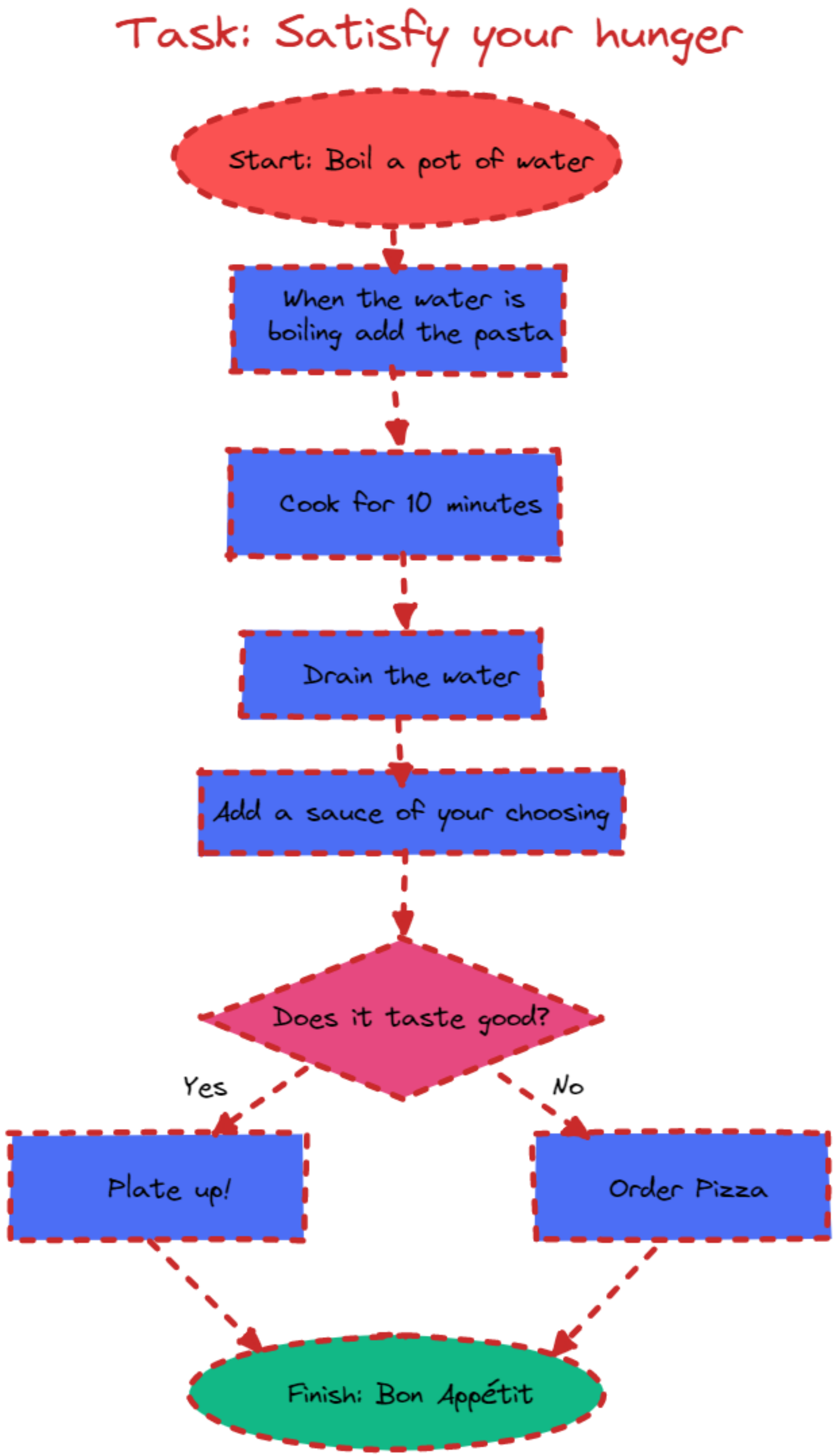
- Abstraction involves **reducing complexity by focusing on the main idea** and ignoring specific details that are not essential. It simplifies problems by highlighting the core elements.
- **Application in Problem-Solving:**
  - **Example: Designing a mobile app** involves abstraction by focusing on user interface and core functionalities rather than the underlying code initially. The details of the code can be handled later, while the main features are mapped out first.
  - **Real-World Example:** A **city planner** uses abstraction when creating a map of a city. The map includes only essential elements like roads, landmarks, and public transportation routes, omitting less relevant details.
  - **Example:** In **creating a financial report**, abstraction helps by focusing on key financial indicators such as revenue, expenses, and profit margins, rather than every single transaction detail.
  - **Example: Scientists use abstraction** when modeling complex systems like ecosystems. They focus on key species and environmental factors rather than every individual organism and interaction.

Abstraction



Algorithms

- **Definition:** An algorithm is a step-by-step procedure or formula for solving a problem. It provides a clear sequence of actions to be performed.
- **Application in Problem-Solving:**
  - **Example:** When **cooking a recipe**, following the step-by-step instructions is using an algorithm to achieve the desired dish.
  - **Real-World Example:** In **online navigation apps like Google Maps**, algorithms are used to find the shortest path from one location to another by considering various factors such as traffic, road conditions, and distance.
  - **Example:** In **data analysis, sorting algorithms** (like QuickSort or MergeSort) organize data to make it easier to analyze. Sorting sales data by date or amount can help quickly identify trends.
  - **Example:** **Machine learning algorithms** are used to classify data, predict outcomes, and find relationships. For example, a spam filter uses algorithms to determine whether an email is spam based on the content and sender.


Task: Satisfy your hunger
Start: Boil a pot of water
When the water is boiling add the pasta
Cook for 10 minutes
Drain the water
Add a sauce of your choosing
Does it taste good?
Yes
No
Plate up!
Order Pizza
Finish: Bon Appétit

Complexity

- Complexity involves understanding **how different aspects of a problem interact** and the impact of scaling on the problem's solutions. It considers both the intricacy of the problem and the resources required to solve it.
- **Application in Problem-Solving:**
  - **Example:** In **project management**, understanding complexity helps allocate resources and time effectively. Complex projects might need more time and manpower compared to simpler ones.
  - **Real-World Example:** Managing a **large-scale construction project** requires understanding the complexity of coordinating different teams, suppliers, and timelines to ensure the project is completed on time and within budget.
  - **Example:** **Analyzing big data** requires understanding computational complexity to choose the **right algorithms** that can handle large volumes of data efficiently.
  - **Example:** **Predicting customer behavior** in retail requires complex analysis of various factors such as purchase history, browsing behavior, and demographics.

Complexity

Measurement of how an algorithm's performance scales with input size.

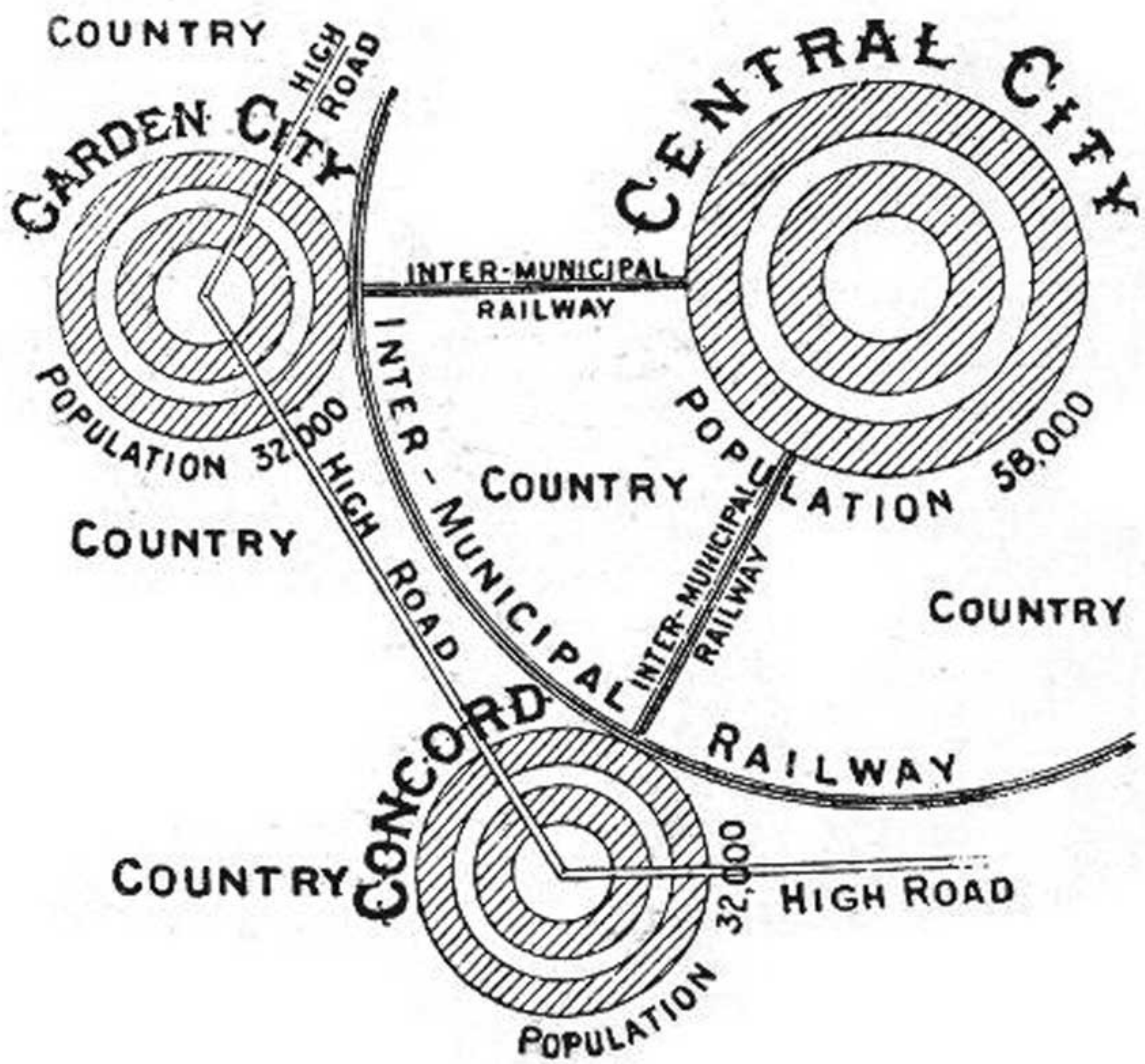
Real-World Example: Sorting a deck of cards. The more cards you have, the more time it takes to sort them. Comparing different methods:

- Sorting by hand (O(n^2))
- Using a sorting machine (O(n log n))

Visual: Graph comparing time complexity (e.g., sorting times for different numbers of cards).

Modeling

- Modeling is **creating a representation of a system** or process to understand and predict its behavior. Models help in simplifying and visualizing complex systems.
- **Application in Problem-Solving:**
  - **Example:** Creating a **business model** to represent how a company generates revenue, incurs costs, and creates value. This helps in planning strategies for growth and profitability.
  - **Real-World Example:** In **urban planning, models of traffic flow** are used to predict and solve congestion problems by simulating different scenarios and their impact on traffic.
  - **Example:** In climate science, modeling is used to **predict future climate conditions** based on current and historical data. Models can simulate how factors like CO2 levels affect global temperatures.
  - **Example:** **Financial models are used to predict market trends** and assess the impact of economic policies or business decisions.



The Garden City movement

Analysis

- Analysis involves **examining data to extract meaningful insights**, identify patterns, and make informed decisions.
- **Application in Problem-Solving:**
  - **Example: Analyzing customer feedback** to improve products and services. By understanding customer needs and preferences, businesses can make data-driven decisions to enhance their offerings.
  - **Real-World Example:** In healthcare, **analyzing patient data helps doctors diagnose** conditions, track health outcomes, and tailor treatments to individual needs.
  - **Example:** Performing **statistical analysis to determine the effectiveness of a new marketing campaign** by comparing sales data before and after the campaign.
  - **Example:** Data analysts use techniques like **regression analysis to understand relationships between variables**, such as the impact of pricing changes on sales volume.

Visualization

- Visualization involves **using graphical representations to present data**, making it easier to understand and communicate insights.
- **Application in Problem-Solving:**
  - **Example:** Using a **Gantt chart** to visualize project timelines, helping project managers track progress and identify potential delays.
  - **Real-World Example:** In **business presentations, using bar charts and pie charts** to convey financial performance or market share makes it easier for stakeholders to grasp complex data.
  - **Example:** Visualizing data through **heat maps** to identify geographic regions with high customer concentration, aiding in targeted marketing efforts.
  - **Example:** Using **line graphs to track trends over time**, such as sales growth, helps in understanding performance and forecasting future trends.

Visualization

TASK

Show Decomposition of the problem: Planning a road trip.

Write the algorithm for the problem: A recipe for baking a cake

Data Encoding and Representation

- Data encoding and representation are crucial for storing, transmitting, and processing different types of data efficiently.
- Data is represented in various formats, including binary, ASCII, Unicode, and specific formats for text, images, audio, and video.

1. Binary Representation

- **Binary is the most fundamental form of data representation in computers. It uses two symbols, 0 and 1, to represent all data. Each binary digit is called a bit, and a group of eight bits forms a byte.**
- **Usage:**
  - Binary is the language of computers. All types of data (text, images, audio, video) are eventually converted into binary for processing by computers.
- **Example:**
  - The binary representation of the decimal number 5 is `101`.
  - The binary representation of the letter 'A' (using ASCII) is `01000001`.

2. Text Encoding

ASCII (American Standard Code for Information Interchange)

- **ASCII is a character encoding standard that uses 7 bits to represent characters. It can encode 128 unique characters, including English letters, digits, and common punctuation marks.**
- **Usage:**
  - Primarily used for encoding text in early computer systems and is still widely used in programming and data formats.
- **Example:**
  - The ASCII code for 'A' is `65` (binary `01000001`).
  - The ASCII code for 'a' is `97` (binary `01100001`).

2. Text Encoding

Unicode

- Unicode is a comprehensive encoding standard that **supports characters from all writing systems in the world**. It uses a **variable-length encoding,** which can range from 8 bits (UTF-8) to 32 bits (UTF-32).

- **Usage:**
  - Unicode is the standard for text representation in modern systems, allowing for the encoding of thousands of characters, including symbols and emojis.
- **Example:**
  - The Unicode for 'A' is `U+0041`.
  - The Unicode for '漢' (a Chinese character) is `U+6F22`.

Text Files and Formats

- **Plain Text (.txt):** Encodes text using basic encoding schemes like ASCII or UTF-8 without formatting.
- **Rich Text Format (.rtf):** Allows text with simple formatting like bold, italics, and different fonts.
- **Markdown (.md):** A lightweight markup language for formatting text, commonly used for documentation.

3. Image Encoding

- **Images are represented as a collection of pixels, with each pixel having a color value. These color values are stored in binary format. Different image formats use different encoding methods to store these values.**

Common Image Formats:

- **JPEG (.jpg, .jpeg):**
  - Uses lossy compression to reduce file size. Suitable for photographs and realistic images. JPEG supports 24-bit color, allowing for over 16 million color combinations.
- **PNG (.png):**
  - Uses lossless compression, preserving image quality. Supports transparency and is ideal for images with text, graphics, and logos.
- **GIF (.gif):**
  - Uses lossless compression and supports up to 256 colors. Commonly used for simple animations.
- **BMP (.bmp):**
  - A bitmap format that stores pixel data without compression. Results in large file sizes.

How Images Are Encoded:

- **Bit Depth:** Determines the number of colors that can be represented in an image. Higher bit depth means more colors and greater image quality.
  - **1-bit:** Black and white images.
  - **8-bit:** 256 colors (common in GIF).
  - **24-bit:** True color (16.7 million colors, common in JPEG and PNG).
- **Color Models:**
  - **RGB (Red, Green, Blue):** Commonly used for digital images, where each color channel can have a value from 0 to 255.
  - **CMYK (Cyan, Magenta, Yellow, Black):** Used for color printing.

4. Audio Encoding

- Audio is represented digitally by sampling sound waves at discrete intervals (sample rate) and encoding these samples into binary format.

Common Audio Formats:

- **MP3 (.mp3):**
  - A **lossy format** that compresses audio by removing parts of sound that are less audible to humans. Widely used for music storage and streaming.
- **WAV (.wav):**
  - A lossless format that stores raw audio data. Offers **high quality but results in large file sizes**.
- **AAC (.aac):**
  - Advanced Audio Coding, used in Apple devices and streaming. Offers better quality than MP3 at similar bit rates.
- **FLAC (.flac):**
  - Free Lossless Audio Codec, **compresses audio without loss of quality.** Suitable for audiophiles and archival.

How Audio Is Encoded:

- **Sample Rate:** The number of samples taken per second, measured in Hertz (Hz). Common sample rates are 44.1 kHz (CD quality) and 48 kHz (professional audio).
- **Bit Depth:** The number of bits used to represent each audio sample. Common bit depths are 16-bit and 24-bit, affecting the dynamic range of the audio.
- **Channels:** Mono (1 channel), Stereo (2 channels), and surround sound (multiple channels).

5. Video Encoding

- Video combines a sequence of images (frames) with audio, represented in binary format. Video encoding compresses these frames to reduce file size while maintaining quality.

Common Video Formats:

- **MP4 (.mp4):**
  - A highly versatile format that supports video, audio, and subtitles. It uses lossy compression and is widely used for streaming and sharing videos.
- **AVI (.avi):**
  - Audio Video Interleave, a format that can contain both compressed and uncompressed video. Offers high quality but results in large file sizes.
- **MKV (.mkv):**
  - Matroska Video, a flexible format that supports multiple audio tracks, subtitles, and chapters. Popular for HD video.
- **MOV (.mov):**
  - A format developed by Apple, commonly used in video editing and supported by QuickTime Player.

How Video Is Encoded:

- **Frame Rate:** The number of frames displayed per second, measured in frames per second (fps). Common frame rates are 24 fps (film), 30 fps (TV), and 60 fps (high-definition video).
- **Resolution:** The number of pixels in each dimension. Common resolutions are 720p (HD), 1080p (Full HD), and 4K (Ultra HD).
- **Compression:** Video codecs like H.264 and H.265 compress video data to reduce file size while maintaining quality. Compression reduces the amount of data needed to represent the video.
- **Bitrate:** The amount of data processed per unit of time, measured in kilobits per second (kbps). Higher bitrates generally result in better quality but larger files.

Summary

Computational thinking concepts are fundamental tools that help us **break down problems, identify patterns, and create effective solutions. In data analysis, these concepts enable us to extract meaningful insights from data, automate processes, and make data-driven decisions.** By applying computational thinking, we can tackle complex challenges more effectively, whether in business, science, or daily life.

Understanding data encoding and representation is essential for efficient data storage, transmission, and processing across different media types. Whether dealing with text, images, audio, or video, **choosing the right encoding and format impacts both the quality and the size of the data, affecting performance and usability** in various applications.