



Computational Thinking

Logika Pemrograman (SIF104)

*Program Studi Informatika – Program Sarjana
Fakultas Teknologi Industri*

Coba hitung $1 + 2 + 3 + \dots + 199 + 200$
selama 2 menit

Dilarang pakai kalkulator/alat bantu hitung lainnya.

Kalau sudah selesai kurang dari 2 menit, angkat tangan

Solusi?

- Kalau hitungnya satu per satu, $1 + 2 + 3 + \dots + 200$ akan terlalu lama
- Bisakah kita lihat polanya?
 - $1 + 200 = 201$
 - $2 + 199 = 201$
 - $3 + 198 = 201$
 - ...
 - $100 + 101 = 201$
- $1 + 2 + 3 + \dots + 198 + 199 + 200 = 201 * 100 = 20100$

$$1 + 2 + 3 + 4 + 5 + 6 + \dots + 200$$

$$1 + 2 = 3$$

$$3 + 3 = 6$$

$$6 + 4 = 10$$

$$10 + 5 = 15$$

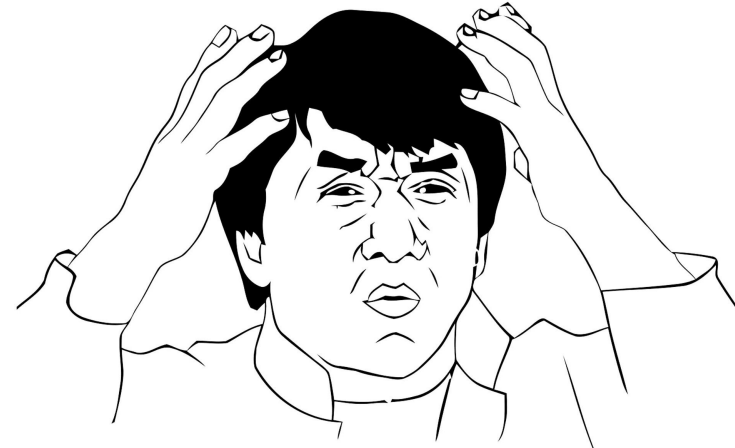
$$15 + 6 = 21$$

...

dan seterusnya sampai ditemukan hasilnya.

Melelahkan & rawan kesalahan?

Pasti ada cara yang lebih baik.



www.coloring-pages.info

$$1 + 2 + 3 + 4 + 5 + 6 + \dots + 200$$

$$1 + 200 = 201$$

$$2 + 199 = 201$$

$$3 + 198 = 201$$

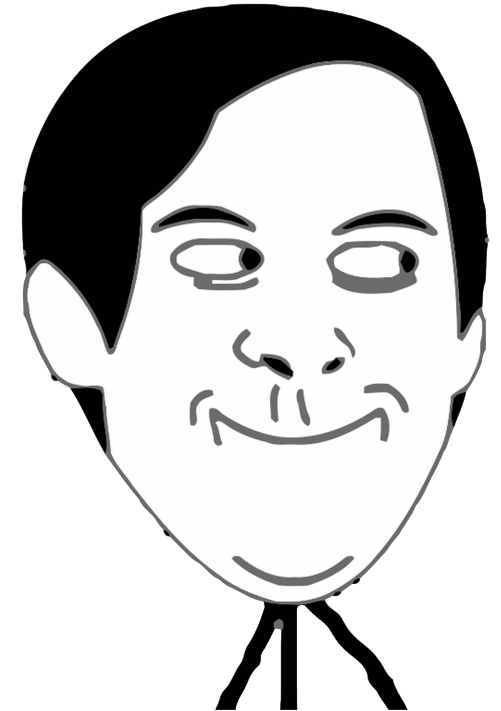
$$4 + 197 = 201$$

...

$$100 + 101 = 201$$

- Bilangan dibagi-bagi menjadi pasangan-pasangan mulai dari tepi lalu bergeser ke tengah.
- Jumlahkan setiap pasangan

Data diinterpretasi jadi
informasi sebagai langkah
menemukan solusi



$$1 + 2 + 3 + 4 + 5 + 6 + \dots + 200$$

$$1 + 200 = 201$$

$$2 + 199 = 201$$

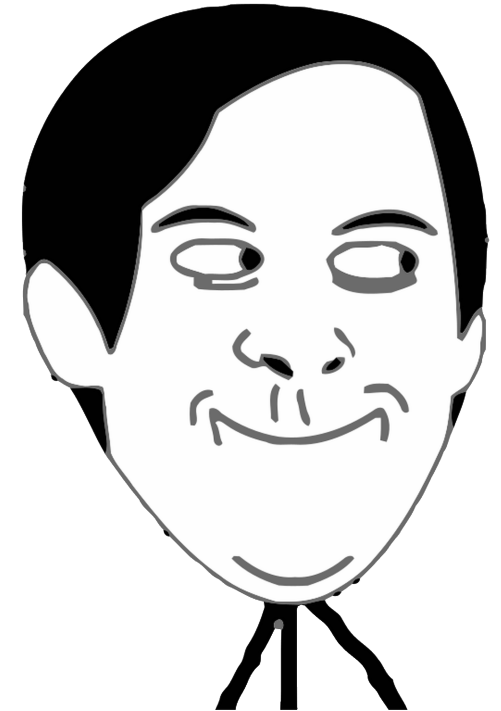
$$3 + 198 = 201$$

$$4 + 197 = 201$$

...

$$100 + 101 = 201$$

Data diinterpretasi jadi
informasi sebagai langkah
menemukan solusi



Hasil penjumlahan setiap pasangan sama dengan 201

$$1 + 2 + 3 + 4 + 5 + 6 + \dots + 200$$

$$1 + 200 = 201$$

$$2 + 199 = 201$$

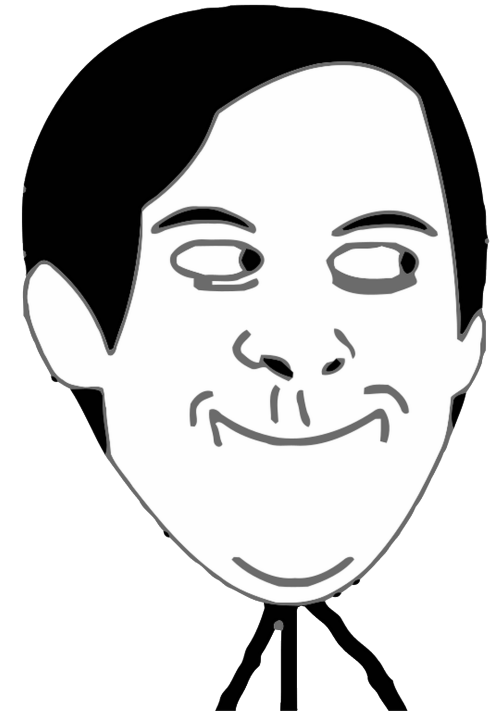
$$3 + 198 = 201$$

$$4 + 197 = 201$$

...

$$100 + 101 = 201$$

Data diinterpretasi jadi
informasi sebagai langkah
menemukan solusi



- Ada **100 pasang bilangan** yang **jumlahnya sama**
- **Karena jumlahnya sama**, hasil akhir dapat diperoleh dari mengabaikan pasangan kedua dan seterusnya. Cukup **pasangan pertama dikali 100** = 20100

$$1 + 2 + 3 + \dots + n$$

Penjumlahan 1 sampai 200 dapat dihitung dengan:

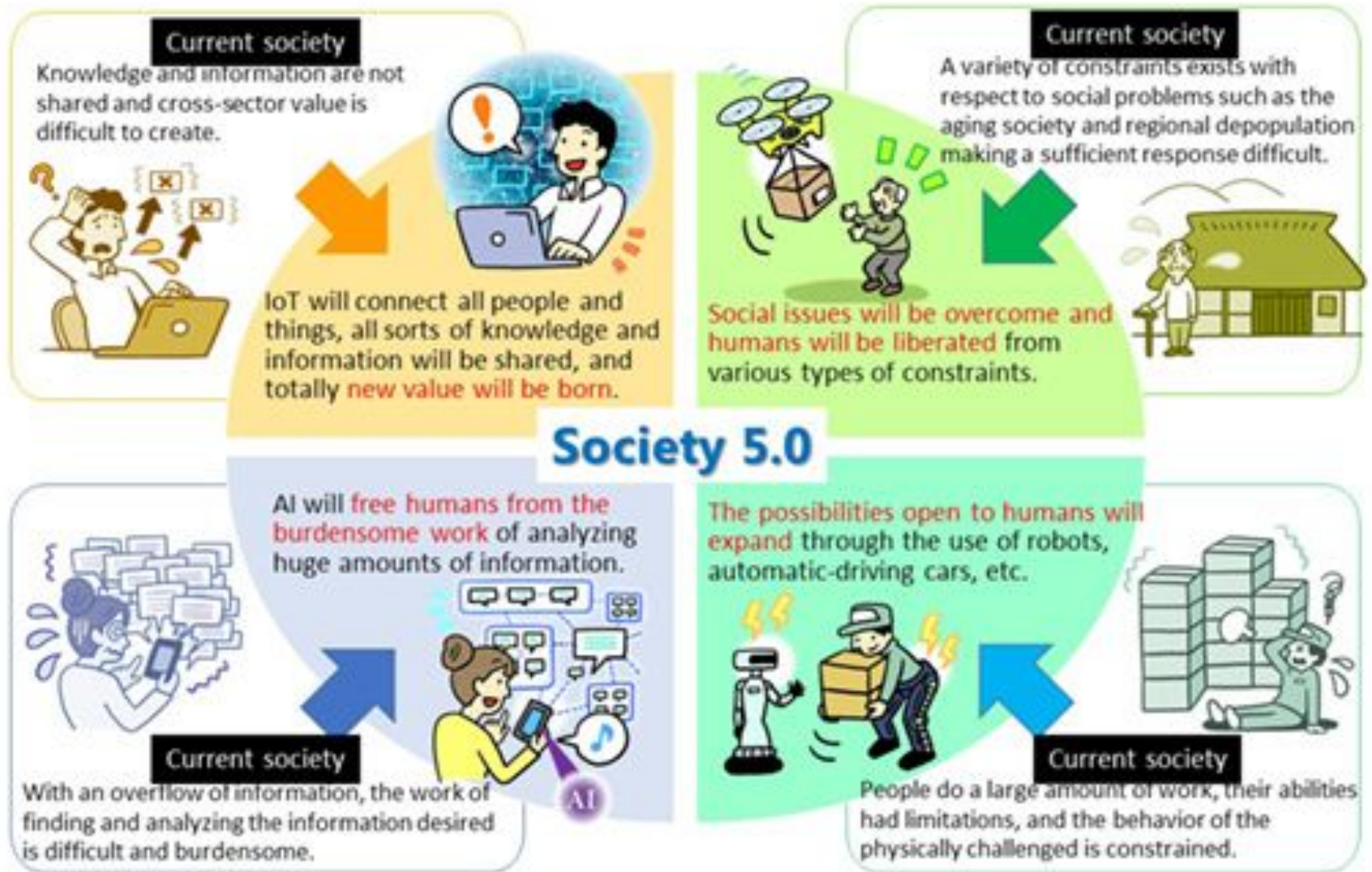
1. $1 + 200 = 201$
2. $200/2 = 100$
3. $201 \times 100 = 20100$



Penjumlahan 1 sampai 200 dapat dihitung dengan:

1. Jumlahkan $1 + n$
2. Bagi n dengan 2
3. Kalikan hasil nomor 1 dengan hasil nomor 2

Latar Belakang



[source: CAO, Japan]

Latar Belakang



Volatility, uncertainty, complexity and ambiguity

https://en.wikipedia.org/wiki/Volatility,_uncertainty,_complexity_and_ambiguity

Top 10 Skills

2015

1. Complex Problem Solving
2. Coordinating with Others
3. People Management
4. **Critical Thinking**
5. Negotiation
6. Quality Control
7. Service Orientation
8. Judgement and Decision Making
9. Active Listening
10. **Creativity**

2025

1. **Analytical thinking**
2. **Creative thinking**
3. **Resilience, flexibility and agility**
4. **Motivation and self-awareness**
5. **Curiosity and lifelong learning**
6. **Technological literacy**
7. **Dependability and attention to detail**
8. **Empathy and active listening**
9. **Leadership and social influence**
10. **Quality control**

4C's 21st Century Learning



OPINION | 21ST CENTURY SKILLS

The 5th 'C' of 21st Century Skills? Try Computational Thinking (Not Coding)

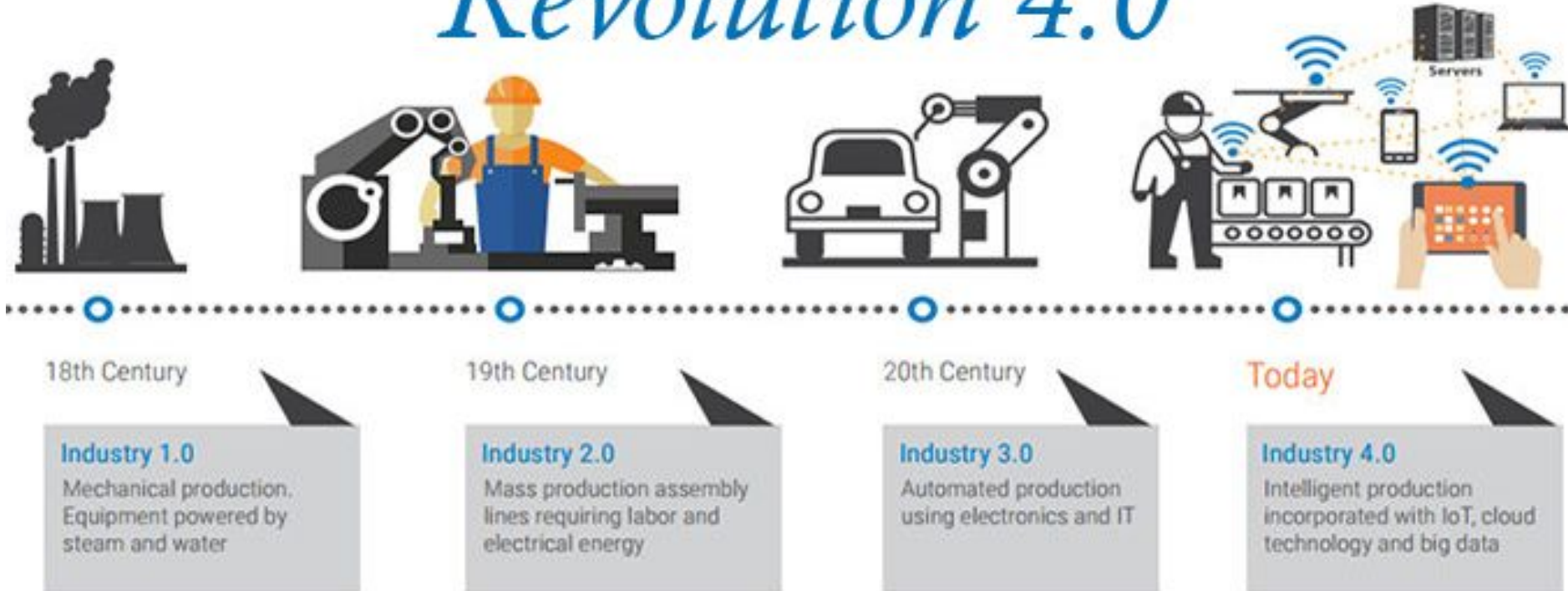
By Shuchi Grover Feb 25, 2018



<https://www.edsurge.com/news/2018-02-25-the-5th-c-of-21st-century-skills-try-computational-thinking-not-coding>

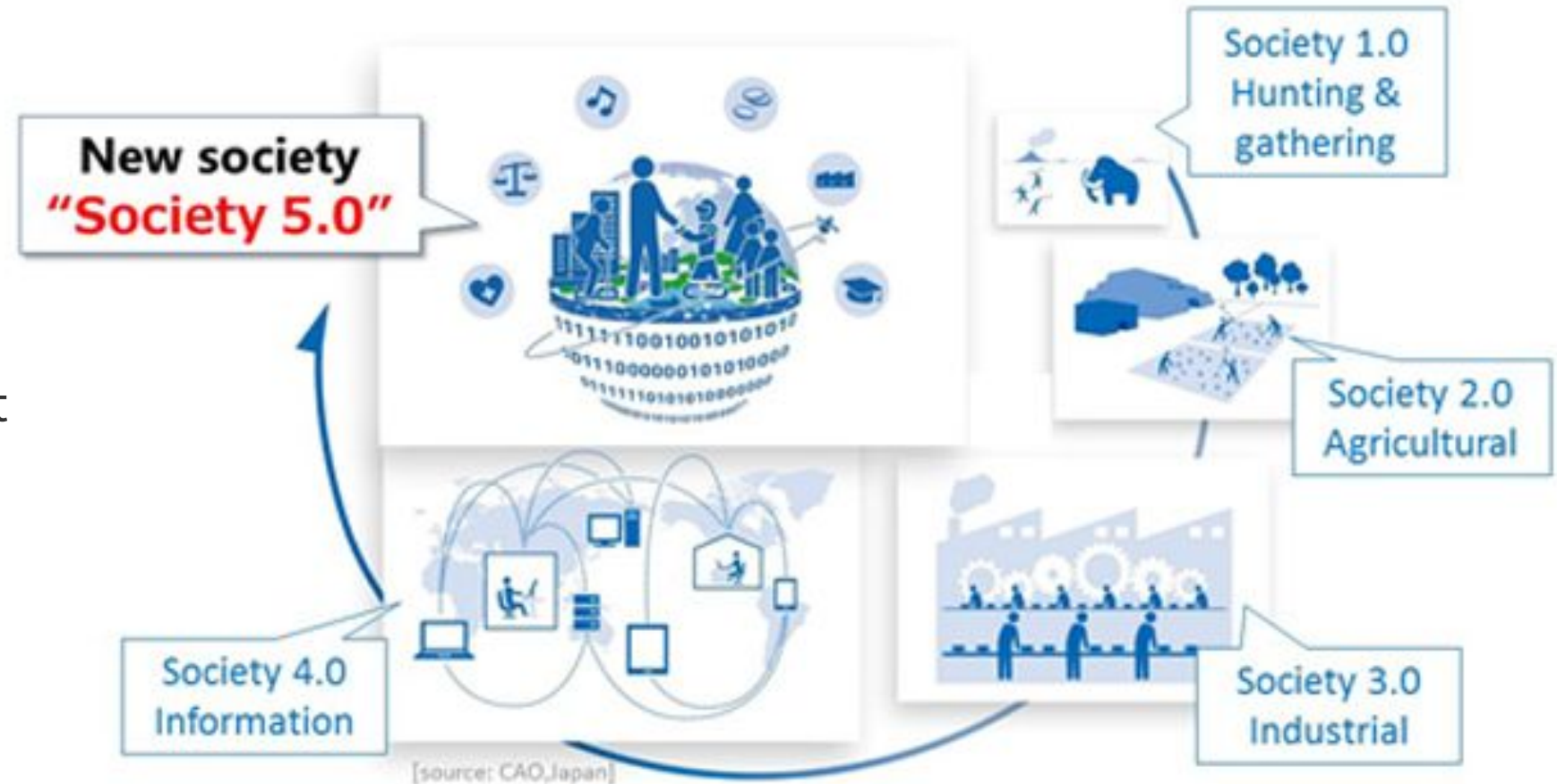
The 5th “C”

The Industrial Revolution 4.0



The 5th “C”

"A human-centered society that balances economic advancement with the **resolution of social problems by a system** that highly **integrates cyberspace and physical space.**"



https://www8.cao.go.jp/cstp/english/society5_0/index.html

"Computational Thinking is the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent."

J. Cuny, L. Snyder, and J. M. Wing.
Demystifying Computational Thinking
for Non-Computer Scientists, 2010

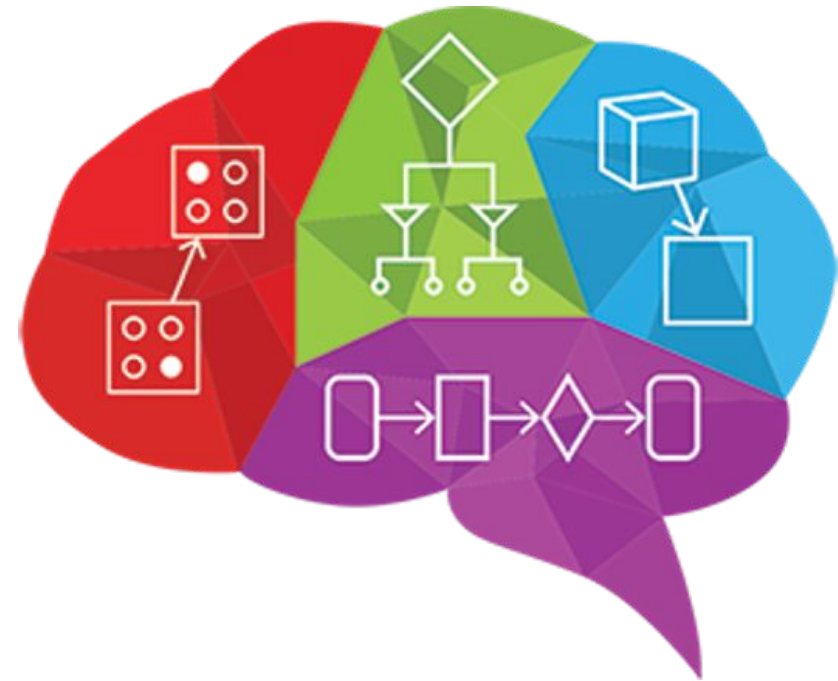


Carnegie Mellon University
Pennsylvania, US

Computational Thinking

Dua aspek penting dari definisi **Jeannette Wing**:

- **Computational thinking** adalah sebuah proses pemikiran, yang terlepas dari teknologi.
- **Computational thinking** adalah *metode penyelesaian masalah* yang dirancang untuk dapat selesai dan dijalankan oleh *manusia, komputer atau kedua-duanya*.

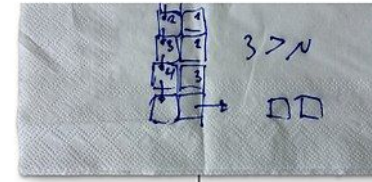


Computational Thinking

“Computational thinking (CT) is a set of *problem-solving* methods that involve *expressing problems* and *their solutions* in ways that a *computer could also execute*.”

Abstraction

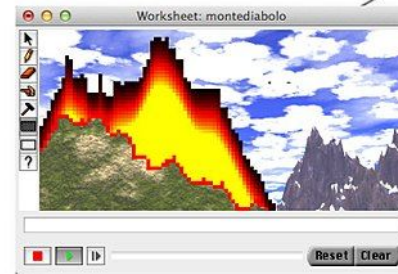
Problem Formulation



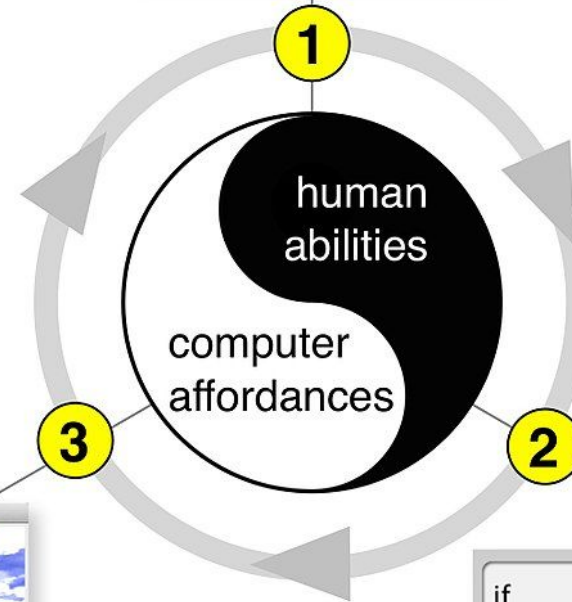
"how does a mudslide work?"

Analysis

Solution Execution and Evaluation



visualize the consequence of thinking



Automation

Solution Expression



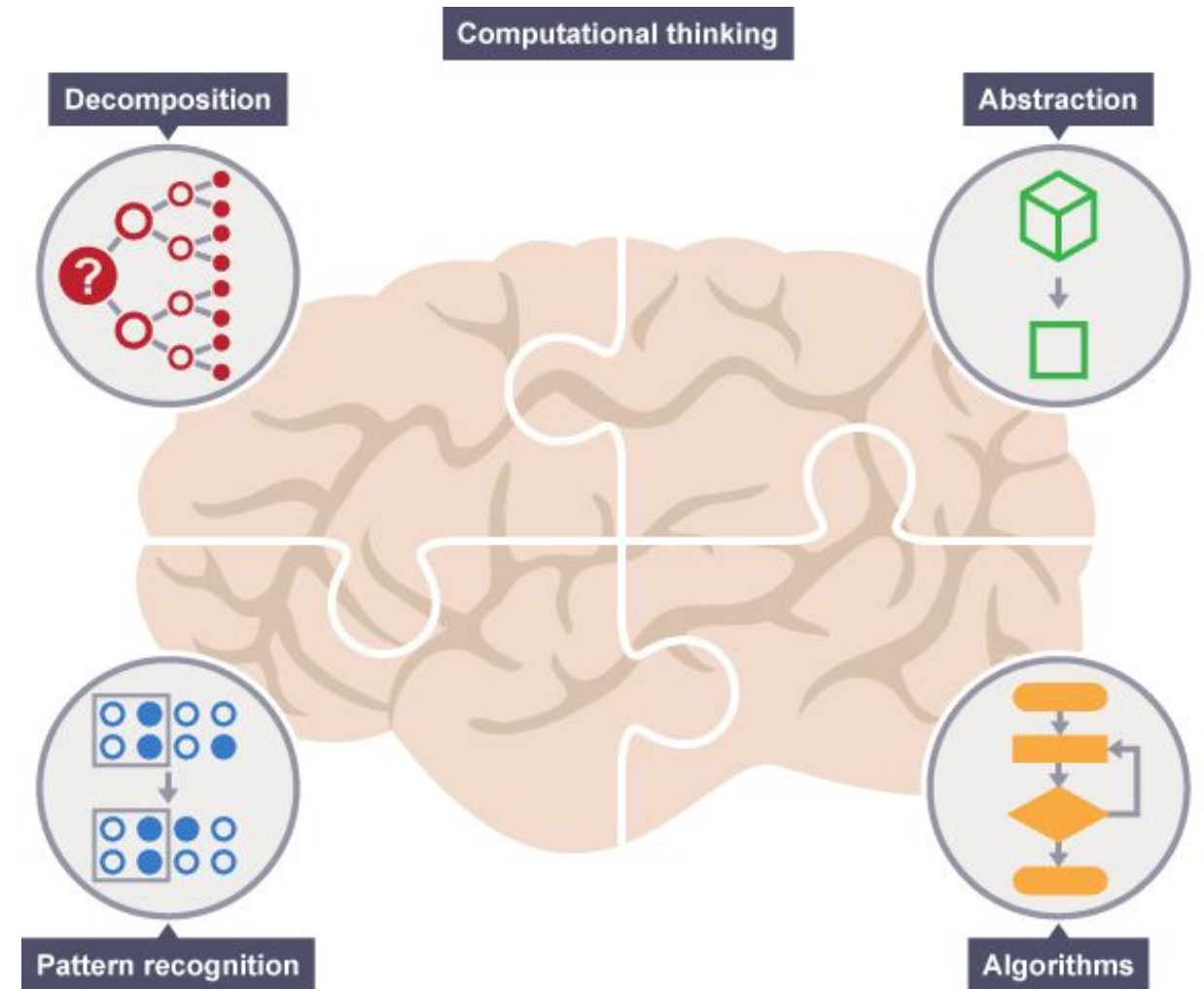
build simple model of gravity

https://en.wikipedia.org/wiki/Computational_thinking

Computational Thinking

- **Decomposition**– breaking down a complex problem or system into smaller, more manageable parts
- **Pattern recognition**– looking for similarities among and within problems
- **Abstraction**– focusing on the important information only, ignoring irrelevant detail
- **Algorithms**– developing a step-by-step solution to the problem, or the rules to follow to solve the problem”

<https://www.bbc.co.uk/bitesize/topics/z7tp34j>



Computational Thinking - Definisi Operasional

Proses penyelesaian masalah yang mencakup karakteristik berikut:

- **Merumuskan masalah** dengan cara yang memungkinkan manusia dapat menggunakan komputer dan alat lainnya untuk membantu menyelesaikannya
- **Mengorganisasi dan menganalisis** data secara logis
- **Merepresentasikan data** melalui abstraksi, seperti model dan simulasi

Computational Thinking - Definisi Operasional

- **Mengotomatiskan solusi** melalui proses berpikir secara algoritmik (dalam serangkaian langkah terurut)
- **Mengidentifikasi, menganalisis, dan menerapkan solusi** yang mungkin dengan tujuan *mencapai kombinasi langkah dan sumber daya yang paling efisien dan efektif*
- **Menggeneralisasi dan mentransfer** proses penyelesaian masalah ke berbagai jenis masalah lain
 - Dengan **mengidentifikasi** pola, kesamaan, hubungan serta mendayagunakan fitur-fitur proses penyelesaian masalah tersebut
 - Ini adalah *salah cara yang cepat menyelesaikan masalah baru yang muncul*

Computational Thinking - Definisi Operasional

Kemampuan **computational thinking** didukung oleh sikap-sikap berikut:

- **Kepercayaan diri** dalam menghadapi **kompleksitas**
- **Ketekunan** dalam menangani masalah sulit
- **Toleransi** untuk ambiguitas
- Kemampuan untuk menangani **open ended problems**
- Kemampuan untuk **berkomunikasi** dan **bekerja** dengan orang lain untuk mencapai tujuan atau solusi bersama

$$1 + 200 = 201$$

$$2 + 199 = 201$$

$$3 + 198 = 201$$

$$4 + 197 = 201$$

...

$$100 + 101 = 201$$

1. Dekomposisi

$$1 + 200 = 201$$

$$2 + 199 = 201$$

$$3 + 198 = 201$$

$$4 + 197 = 201$$

...

$$100 + 101 = 201$$

2. Pengenalan Pola

$$1 + 200 = 201$$

$$2 + 199 = 201$$

$$3 + 198 = 201$$

$$4 + 197 = 201$$

...

$$100 + 101 = 201$$

3. Abstraksi

$$1 + 2 + 3 + \dots + n$$

Penjumlahan 1 sampai 200 dapat dihitung dengan:

1. $1 + 200 = 201$
2. $200/2 = 100$
3. $201 \times 100 = 20100$

Data diinterpretasi jadi informasi sebagai langkah menemukan solusi masalah lain yang serupa

Penjumlahan 1 sampai 200 dapat dihitung dengan:

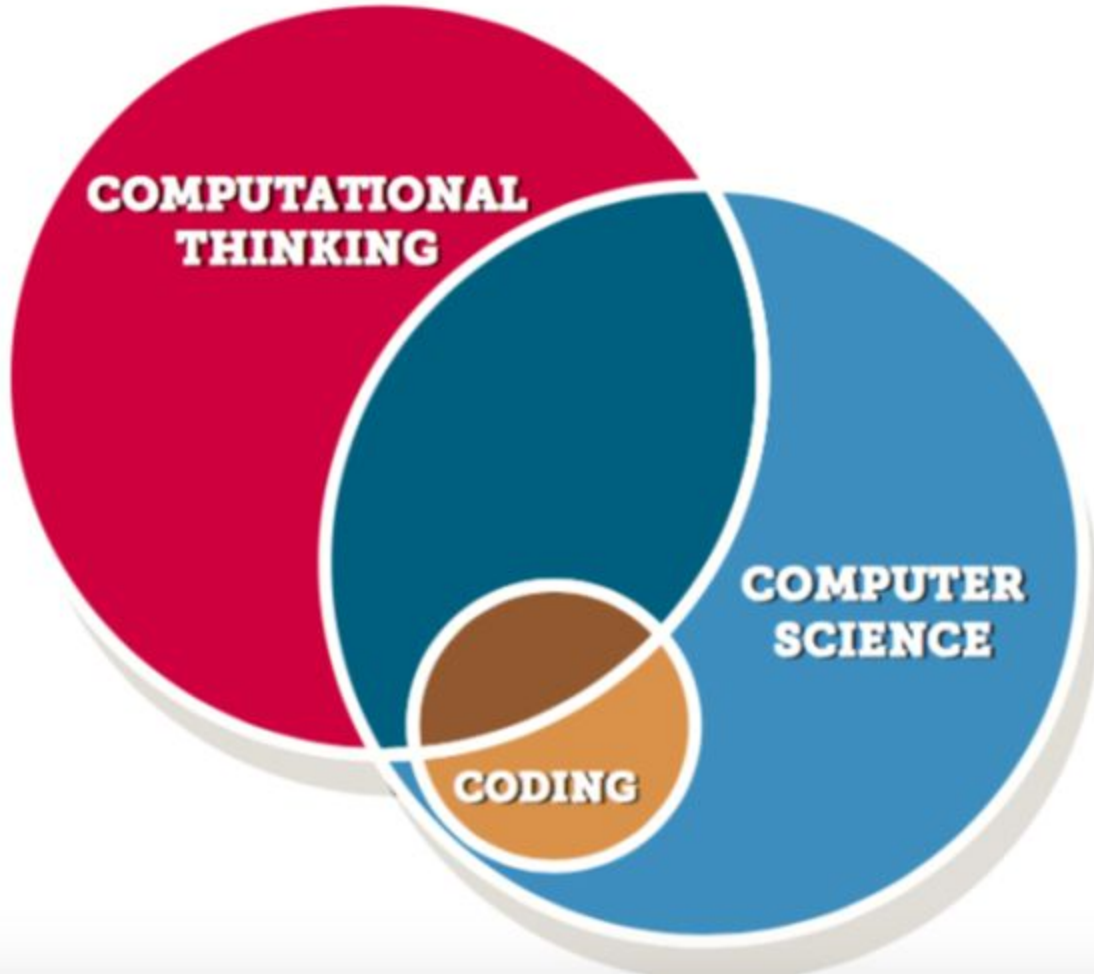
1. Jumlahkan $1 + n$
2. Bagi n dengan 2
3. Kalikan hasil nomor 1 dengan hasil nomor 2

4. Algoritma

Programming vs Computational Thinking

- **Pemrograman** adalah keterampilan yang diperlukan untuk memberitahu komputer apa yang harus dilakukan.
- **Computational thinking** adalah keterampilan untuk belajar dan cara belajar, untuk menciptakan, menemukan, dan memahami dunia (dengan komputer), sebagai ekstensi dan refleksi dari pemikiran diri.
- **Computational thinking** merupakan proses pemikiran di balik pemrograman

Relasi Coding, Computer Science, dan Computational Thinking



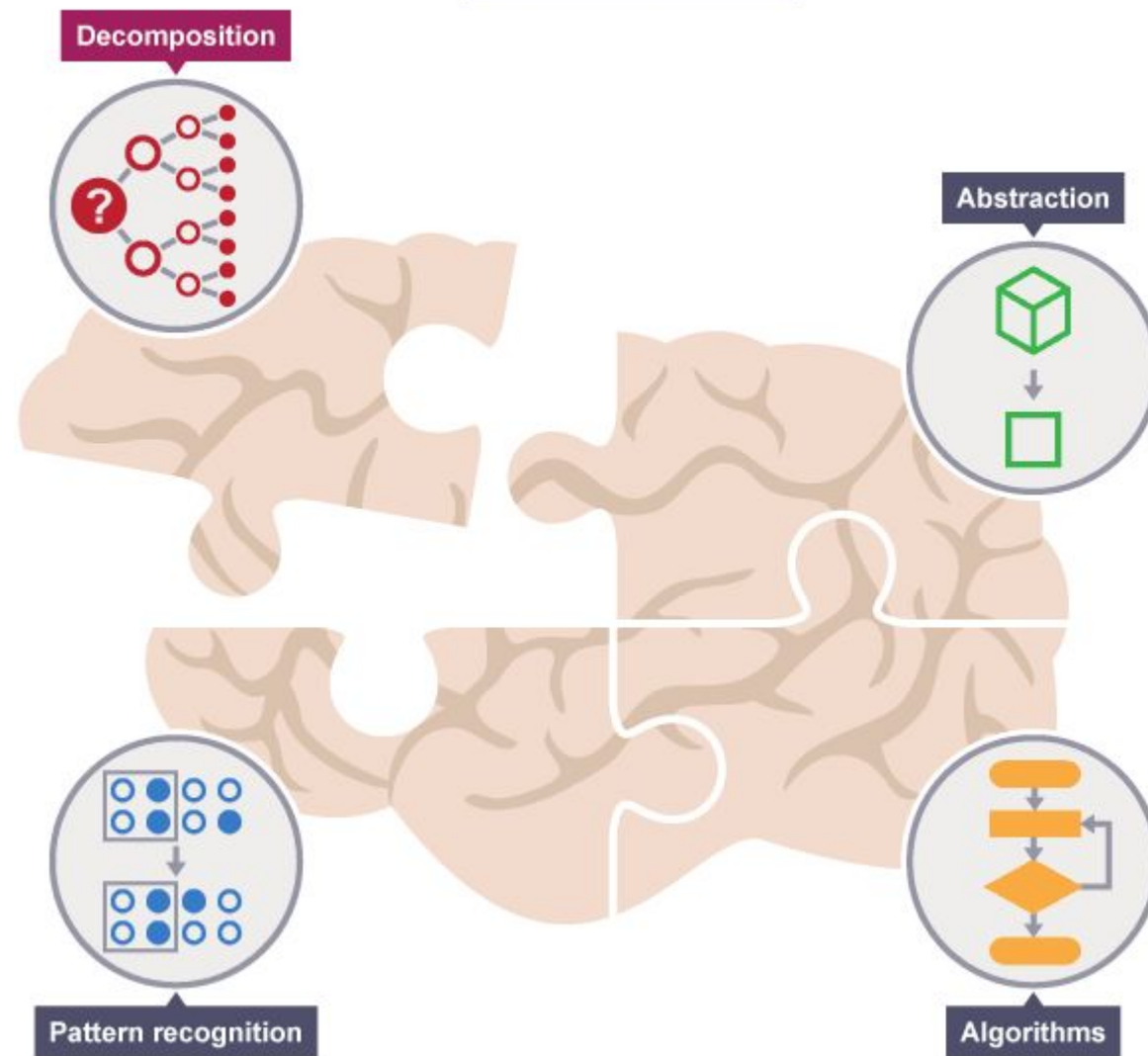
Coding: Kemampuan teknis

CS: Disiplin keilmuan akademis

CT: Kemampuan problem solving, kemampuan dasar semua orang, tidak hanya orang IT

CT #1 Decomposition

Decomposition



Memecah masalah atau sistem yang kompleks menjadi bagian-bagian kecil yang lebih mudah dikelola dan lebih mudah dipahami



Application of Computational Thinking Across the Curriculum!

What equipment do you need for school today?

What were the key events of the 20th Century?

What countries make up Europe?

Write a list of shopping items

Identify the playing positions in a rugby team

Break down a word phonetically

Play Charades!

Break the story of Romeo and Juliet down into it's main sections

Decomposition

Breaking something down into smaller parts

Primary Terminology – “Break Apart”

Early Years Foundation Terminology – “Pieces”

Identify the instruments used within a song

Top trumps!

How does your post get to you?

Break down a typical day at school

Identify the different parts of a bike What components make up the wheel?

Build a computer game using scratch. Think about graphics, levels, character...

Pack a bag for your summer holiday

Write down a list of ingredients for a Victoria sponge cake

What characters will you create for your story?

What information do you need to work out the circumference of a circle?

How did Wales do in Euro 2016?

Explain the movements for the different pieces in a game of chess

Contoh: Bagaimana sepeda bekerja?



Contoh: Menyikat gigi

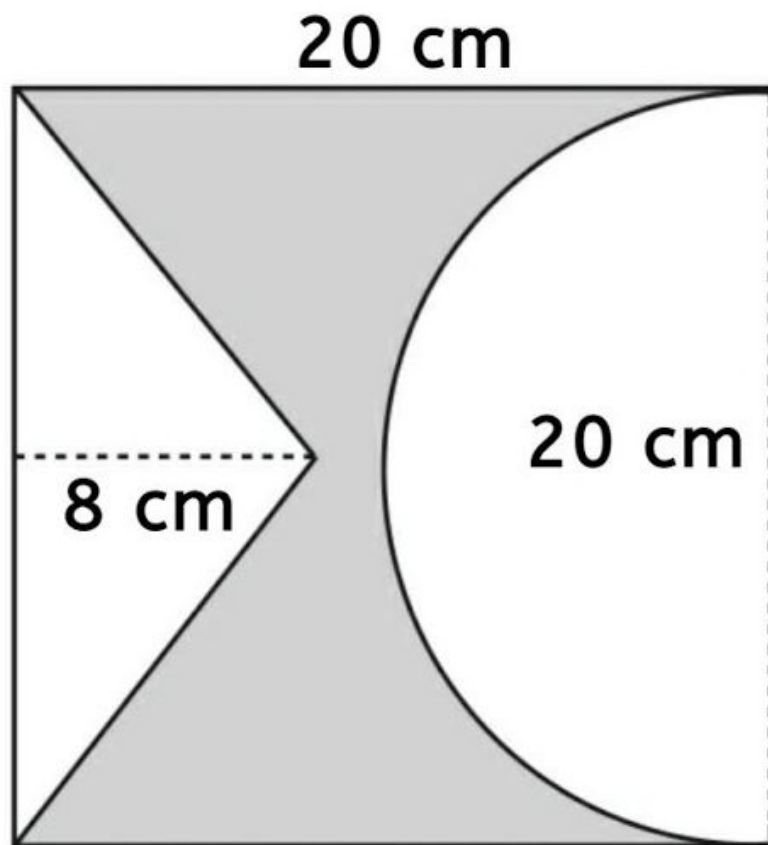
- Sikat gigi mana yang dipakai?
- Berapa lama menyikat gigi?
- Bagaimana tingkat penekanan bulu sikat gigi terhadap gigi?
- Pasta gigi apa yang dipakai?

Contoh: *Anda ingin menonton film di bioskop*

- Apa saja masalah yang perlu diperhatikan?

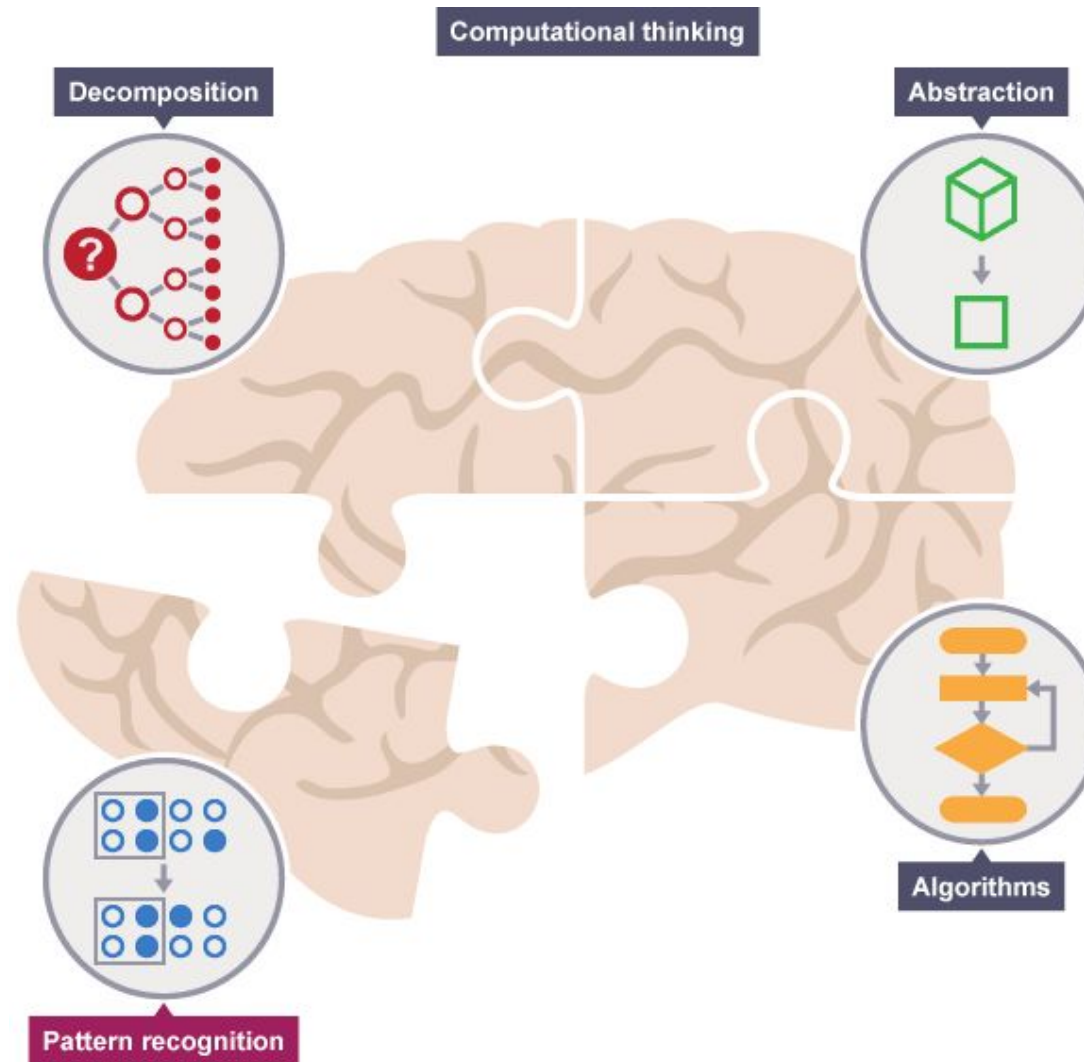


Contoh: Bagaimana menghitung luas daerah yang diarsir?



CT #2 Pattern Recognition

Pattern Recognition



Menemukan **kesamaan** atau **pola** di antara **masalah kecil** yang dapat membantu kita memecahkan masalah yang lebih kompleks dengan lebih **efisien**



Application of Computational Thinking Across the Curriculum!

Pattern Recognition

Looking for similarities or trends

Primary Terminology – “Patterns”
Early Years Foundation Terminology – “Matching”

“That word sounds like...”

Critically review an existing piece of work.

Give feedback comparing work to specific criteria

Correct application of Male/Female tenses

Logic puzzles

Preferred playing positions in a sport

Pattern and sequence matching

Spot the difference

Days of the week/Month

Code breaking

“What tactics worked well the last time we played them?”

Maths patterns, e.g. Fibonacci series (1,1,2,3,5,8,13,21...)

What drawing technique would be best to use for that style of image?

Fit shapes into correctly shaped holes

Sorting and classifying activities

Word search

Chess tactics

Solitaire

Sudoku

Does the star wars and superman theme tunes sound the same?

What do platform computer games have in common?

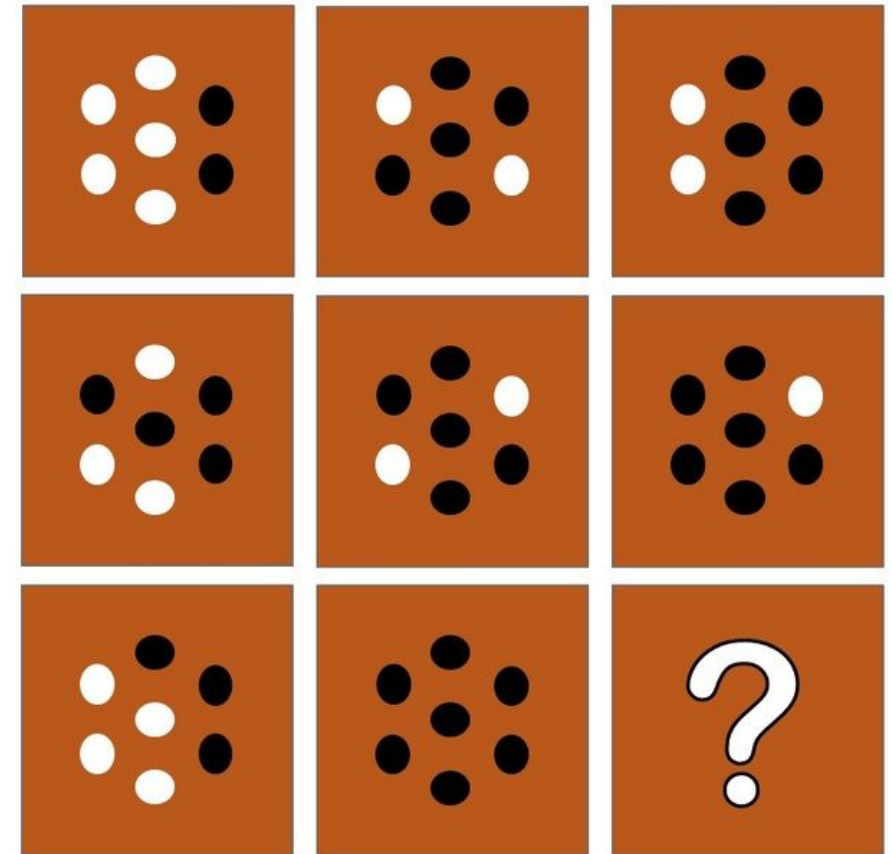
Identify gradients /contours in an OS map to measure steepest route

Times tables

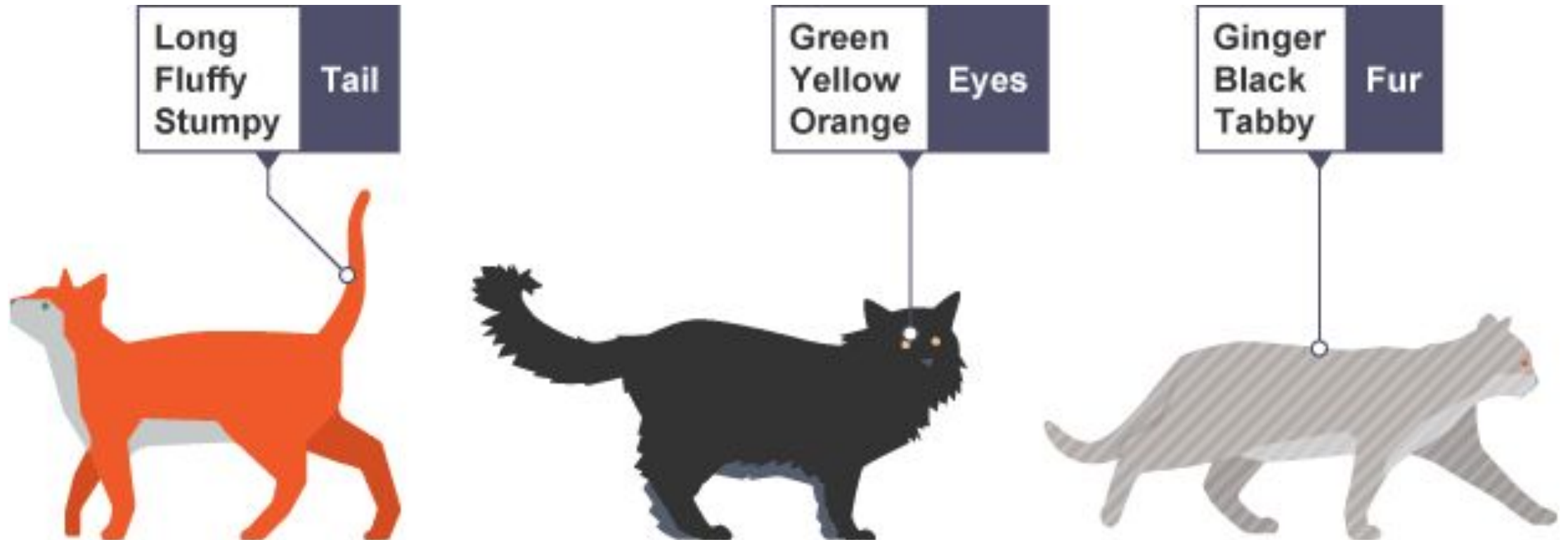
Identify similar rifts within a song

Kenapa perlu mencari pola?

- Tugas menjadi lebih mudah karena kita bisa menggunakan **solusi yang sama** untuk **masalah yang mirip**
- Semakin banyak pola yang kita temukan, semakin mudah dan cepat kita menyelesaikan masalah



Contoh: Menemukan kesamaan pada kucing



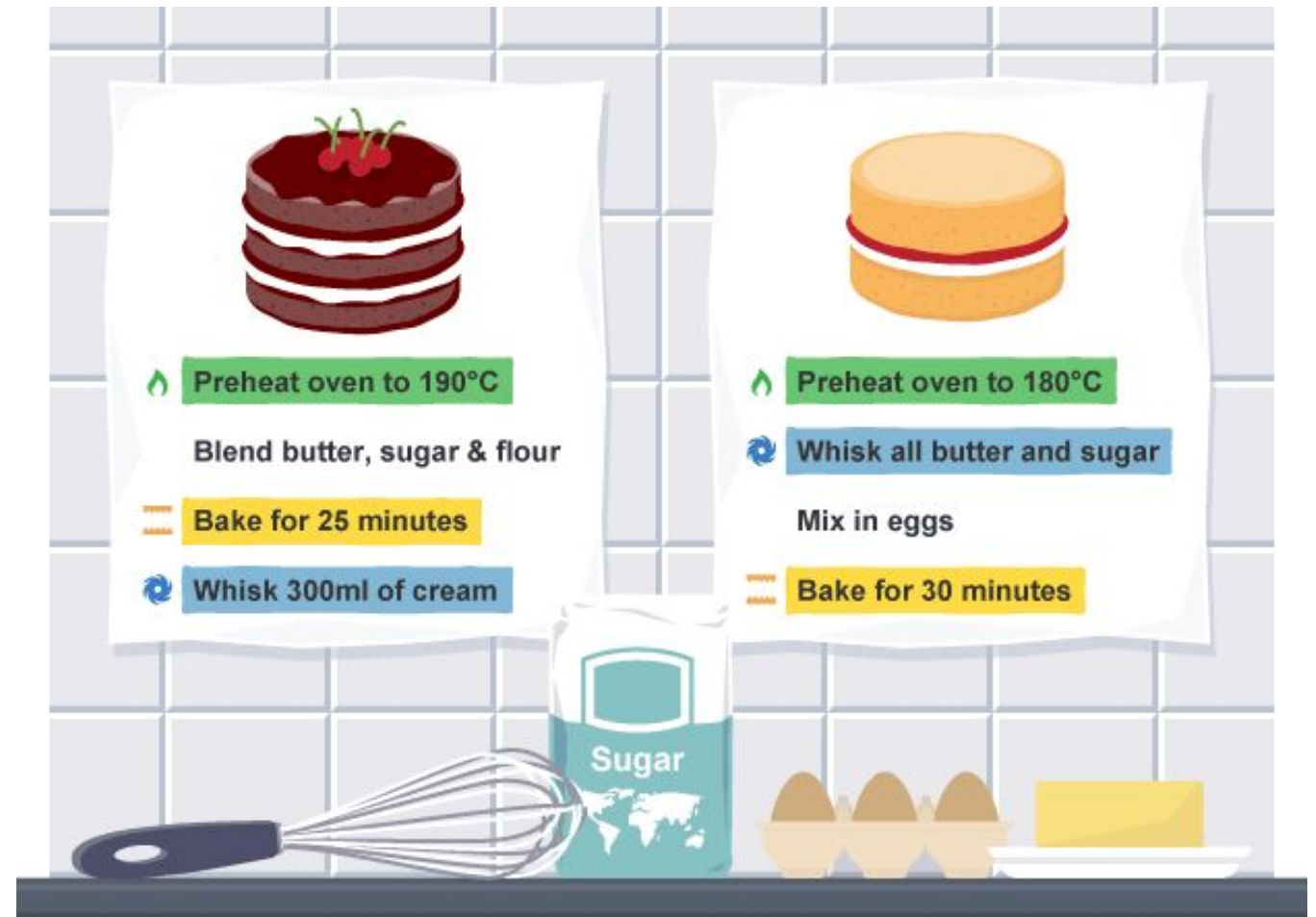
Contoh: Pola Angka

1, 4, 7, 10, 13, 16, 19, 22, 25, ...

3, 8, 13, 18, 23, 28, 33, 38, ...

25, 23, 21, 19, 17, 15, ...

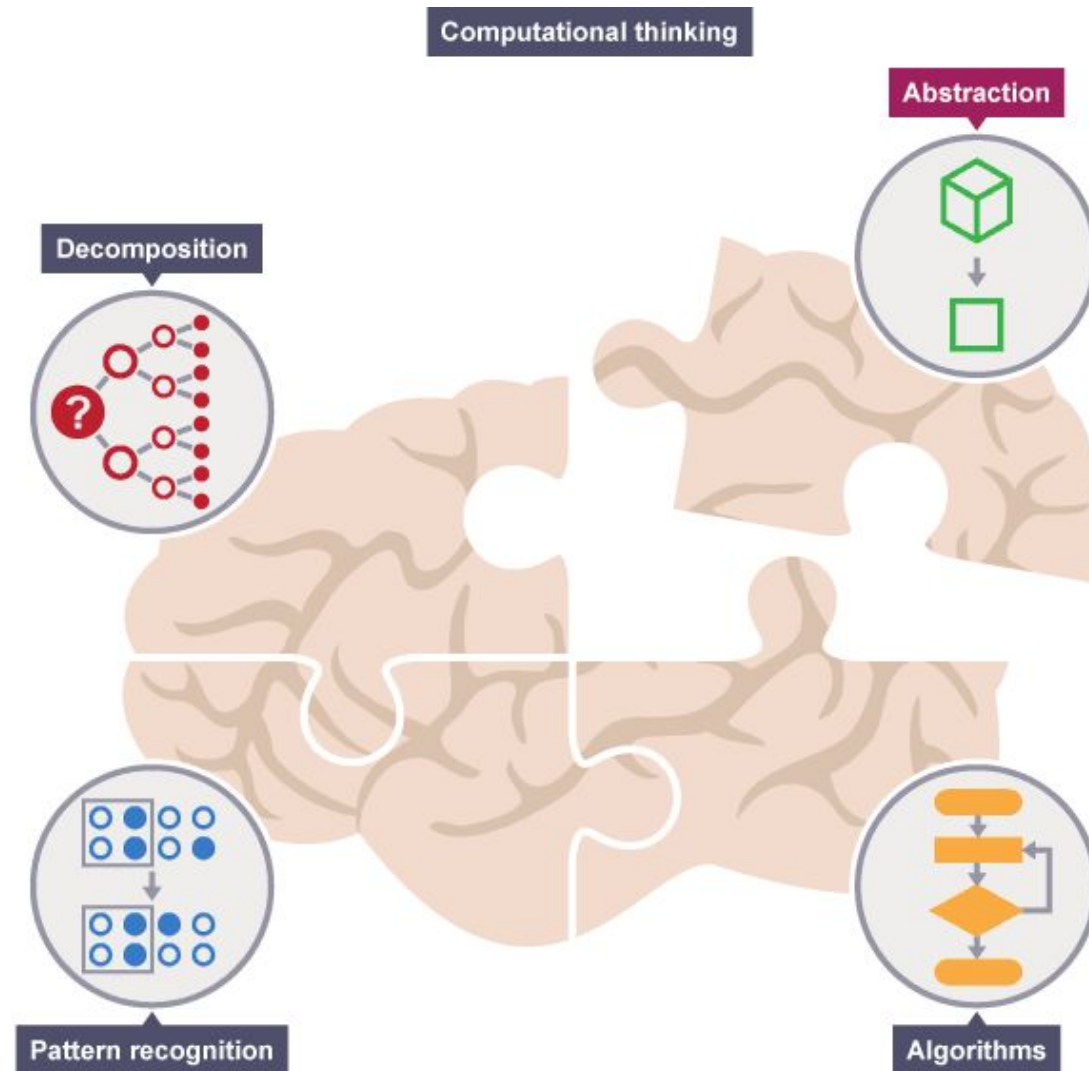
Contoh: Pola membuat kue



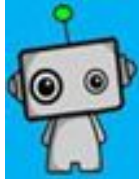
- Preheat oven to *temperature*
- Whisk *quantity of ingredients*
- Bake for *time*



CT #3 *Abstraction*



Mengabaikan karakteristik yang **tidak kita perlukan** untuk **berkonsentrasi** pada **hal penting**, lalu **membuat representasi** (ide) dari apa yang coba kita selesaikan



Application of Computational Thinking Across the Curriculum!

"What did you learn in today's lesson?"

Write a long term training plan

Write a blurb for your movie

Crosswords

Draw a picture of your family

What noise does a dog make?

Following a subway map

Write a match report

Create a model

What are the key skills you need to be a good Hockey player?

Draw a cartoon

Write a synopsis of your story

Explain your idea in 30 seconds!

Articulate

Draw a concept design

What are the key calculations required within the formula?

"What's the weather forecast for today?"

Charades

What happened to Henry VIII's wives?

Recreate the image

Explain briefly what will happen in your computer game.

Recreate the Eiffel Tower in Mine craft!

Who is David Beckham?

Demonstrate the technique used to ...

Do an impression of...

"What's your plan for the weekend?"

"How does that song go again?"

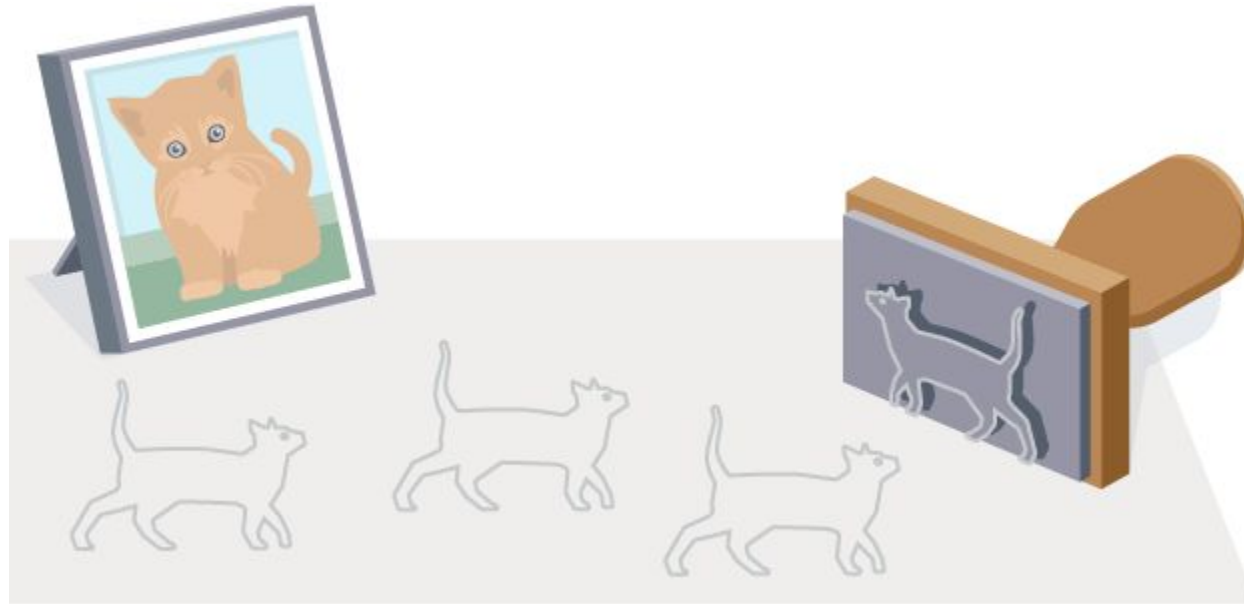
Abstraction

Focusing on what's important,
ignoring what is unnecessary

Primary Terminology – "Thoughts"

Early Years Foundation Terminology – "Ideas"

Contoh: Abstraksi ketika menggambar kucing



YES

Perlu diketahui:

dua mata, dua telinga,
empat kaki, punya ekor

Tidak perlu tahu:

suara kucing,
makanan kesukaan



NO

Kenapa abstraksi itu penting?

- Abstraksi memungkinkan kita untuk *membuat gagasan umum* terkait masalah dan cara penyelesaiannya.
- Abstraksi **menghapus** detail tertentu ataupun **pola yang tidak membantu** menyelesaikan masalah.
- Abstraksi membantu dalam **pembentukan model**
- Jika tidak melakukan abstraksi, solusi ***mungkin bisa jadi salah.***
Contoh kasus menggambar kucing: jika **tidak** melakukan abstraksi, kita ***mungkin berpikir bahwa semua kucing matanya hijau***

Abstraksi

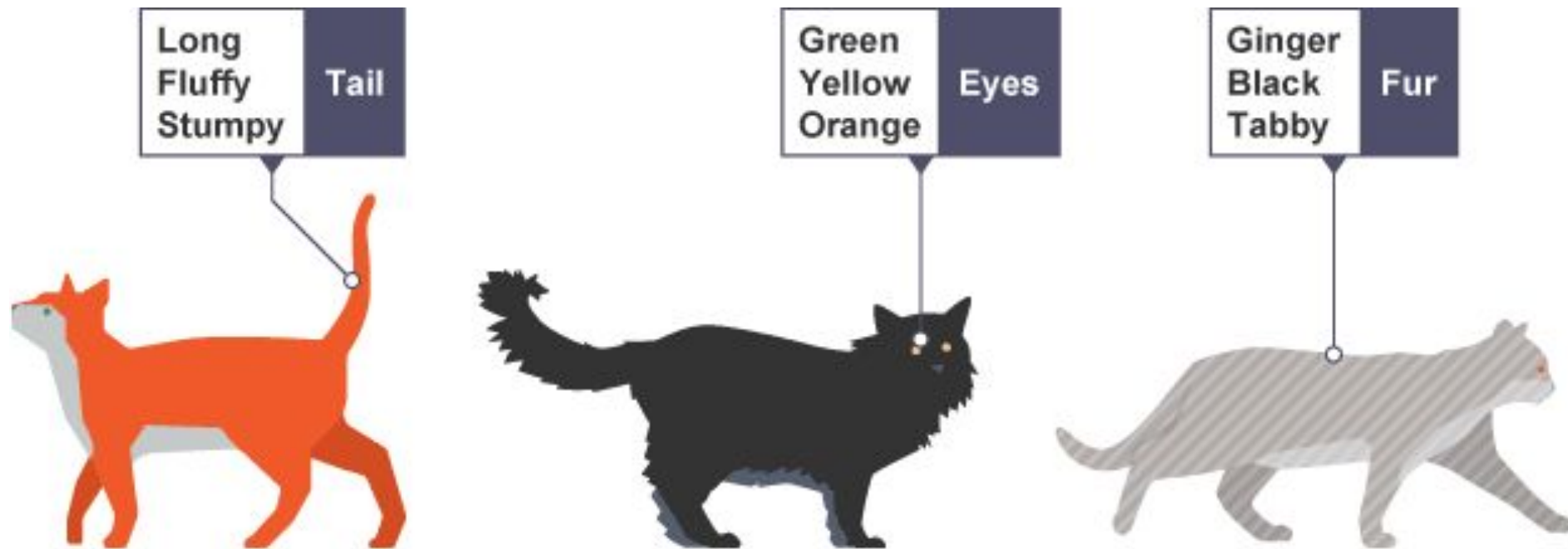
Ketika melakukan abstraksi, kita menghapus detail umum dan mempertahankan pola umum yang relevan.

Contoh kasus: **membuat kue**

Pola umum: kita perlu tahu	Detail spesifik: kita tidak perlu tahu
Kue memiliki komposisi	Komposisinya apa saja
Tiap komposisi ada kuantitas tertentu	Jumlah kuantitas tiap komposisi
Ketika di oven, perlu derajat panas tertentu	Tingkat suhu ketika di oven
Memanggang dalam durasi waktu tertentu	Lama durasi pemangangan

Membuat Model

Model = gagasan umum tentang masalah yang kita coba selesaikan.



Setelah kita memiliki modelnya, kita bisa mendesain **algoritma** pemecahan masalahnya.



Bersambung...