



UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA



**UNIVERSITÀ
DI TRENTO**

Computational Thinking

What's inside the mind of the programmer?

Nice to meet you



- I'm a PhD candidate in Educational Technologies
- My background is in HCI, Cognitive Sciences and Neurosciences
- My research is about CS Education

How brains work is my passion ✨

Which kind of *task* is programming?

let's make it old school and raise your hand 🙋

Think about yourself in the supermarket line:



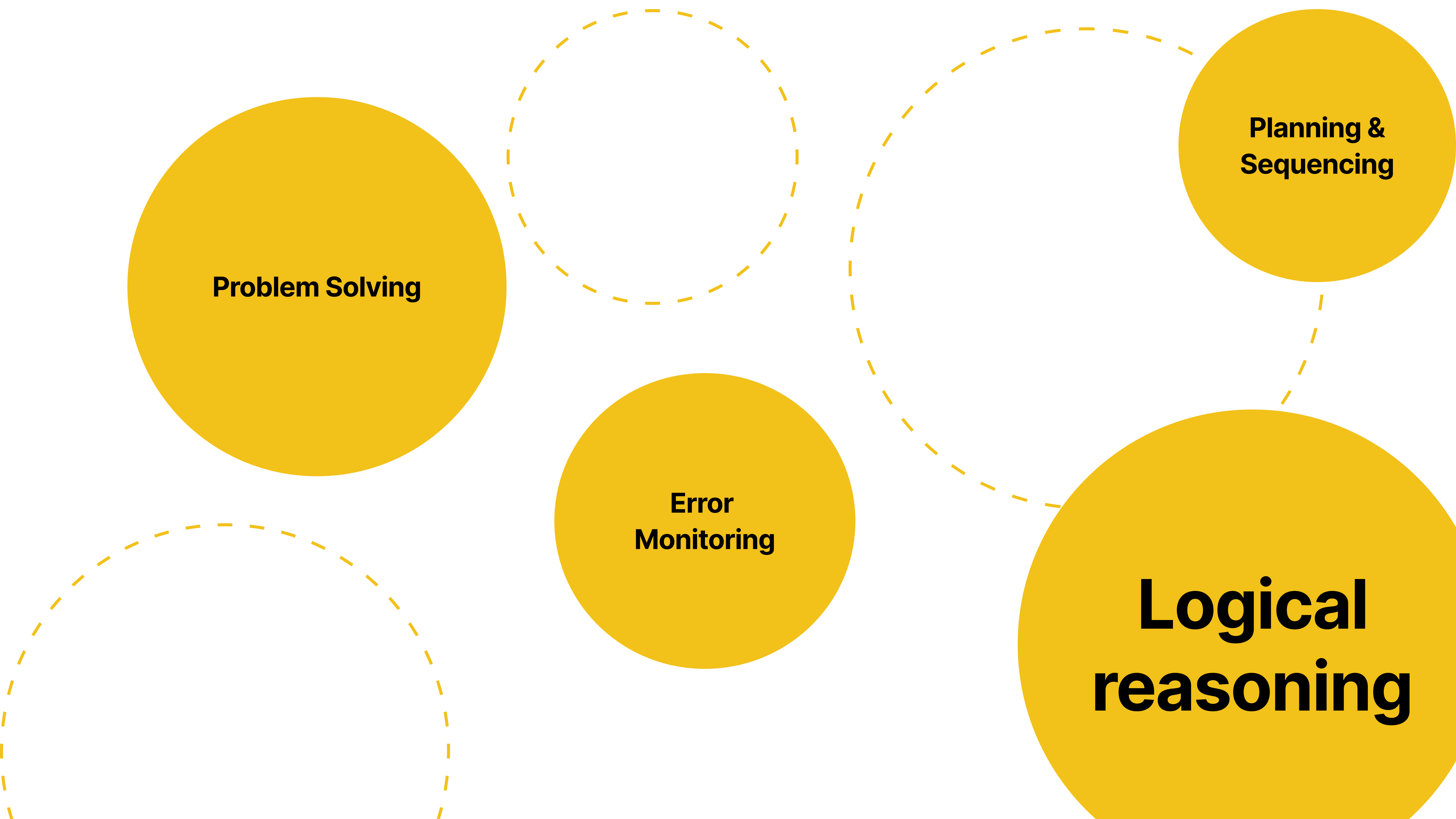
What are you thinking about?

... and why exactly the best way to order your items or upscaling the people in front of you

Which processes and skills are you engaging?

Which kind of *task* is
programming for *your brain*?

Raise your hand again



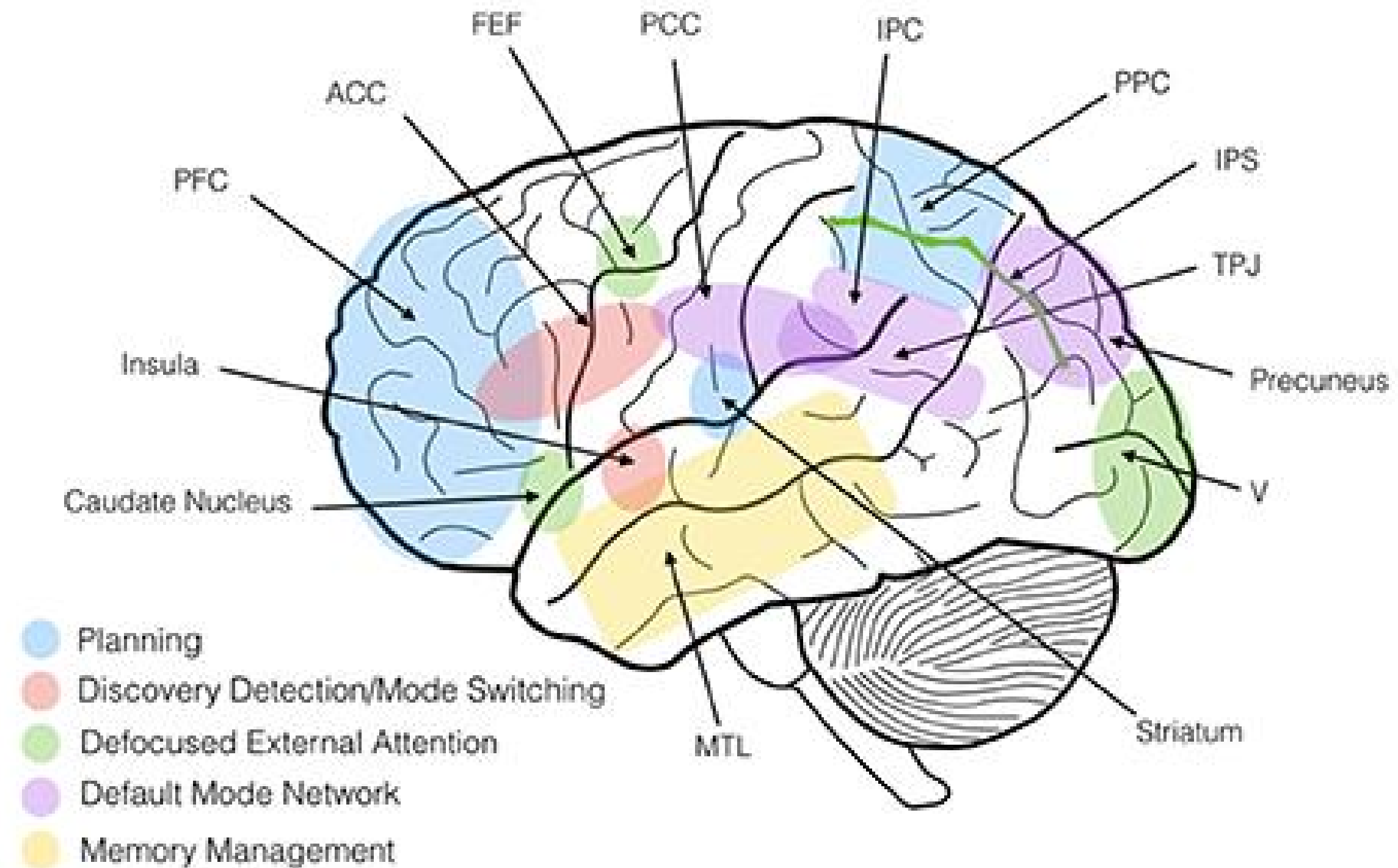
Problem Solving

**Error
Monitoring**

**Logical
reasoning**

**Planning &
Sequencing**

An interplay and parallel of many processes



**Let's do what we do with
any problem**

break it down in pieces

Problem Abstraction and Representation

When you first see the problem and try to figure out a possible solution by visualizing it in your mind.

Solution definition and *expression*

Having an intuition of the possible path is not enough because that needs to become code.

Debugging

If for you it always works nice, but otherwise this is a key step.

Computer programming is a **cognitive *invention***

As for arithmetic, writing and reading, we have some hardware to do so but it's a matter of new developed and installed software that makes the magic happen

Two main perspectives from theory

PROBLEM SOLVING

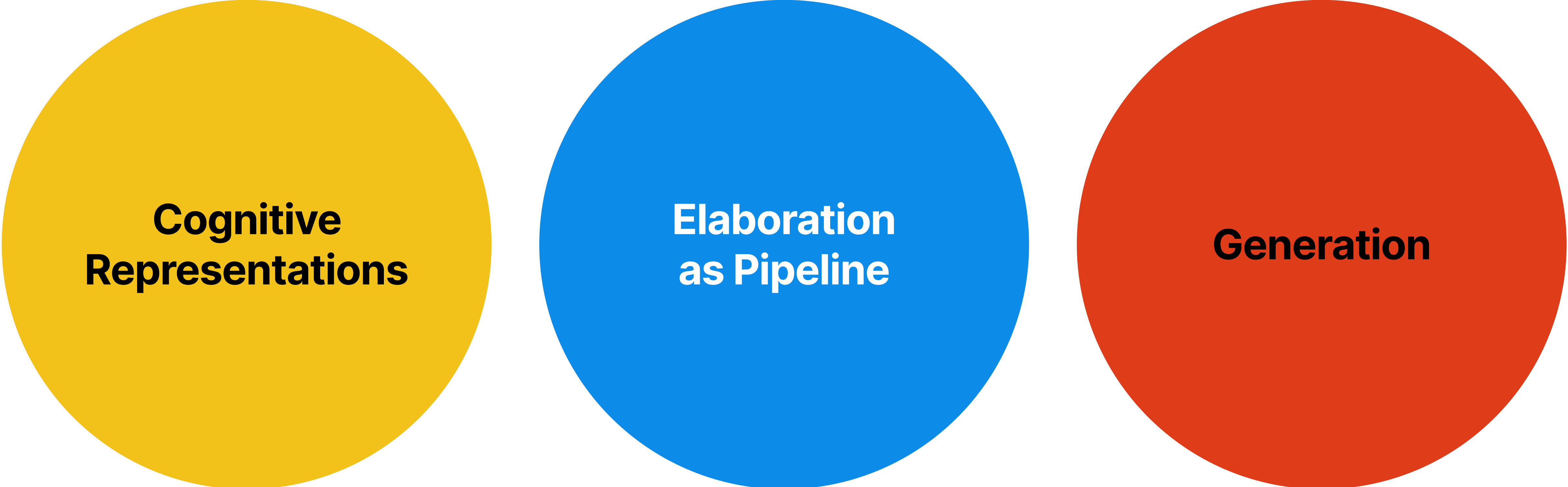
Programming is addressed as a process with **distinct stages**: problem understanding, design, coding, and debugging.

The focus has often been on early **program planning** and so Computer Science has been historically treated as math and problem-oriented, and grouped within the STEM disciplines.

LITERACY

Literacy argues that learning a programming language means acquiring a **symbolic system** to express oneself creatively and communicate. The very term “programming languages” already suggests parallels with natural language and teaching could be aimed at developing language fluency.

Key points

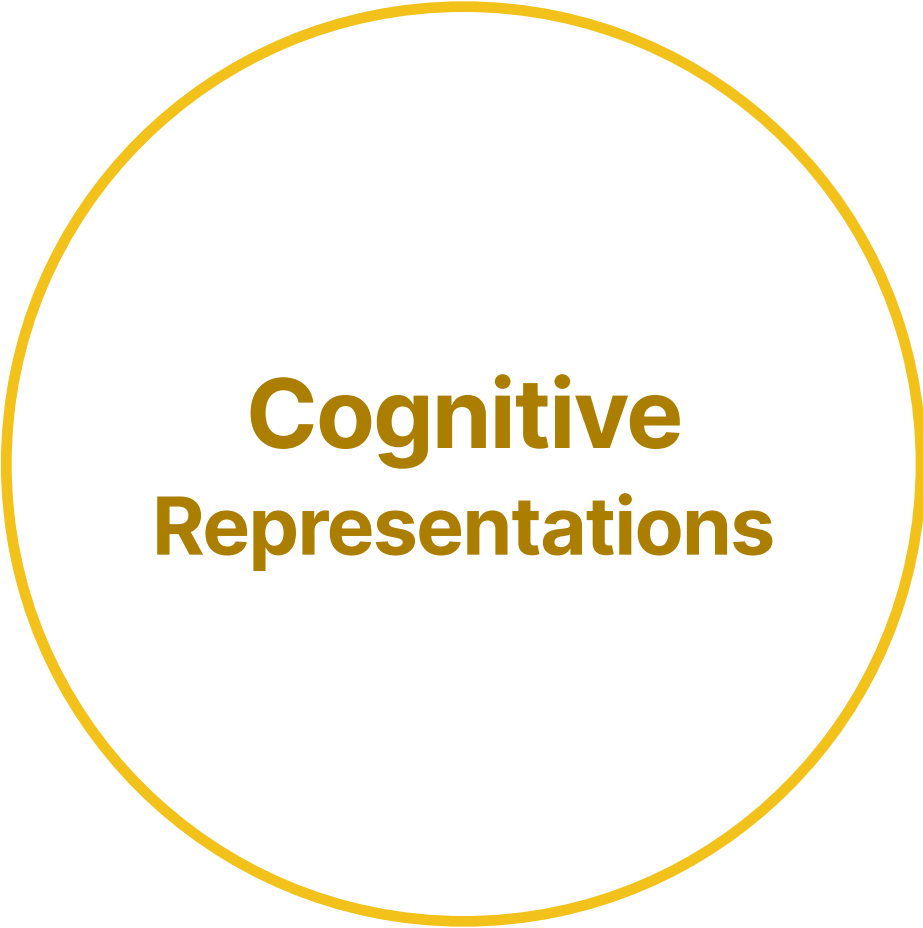


**Cognitive
Representations**

**Elaboration
as Pipeline**

Generation

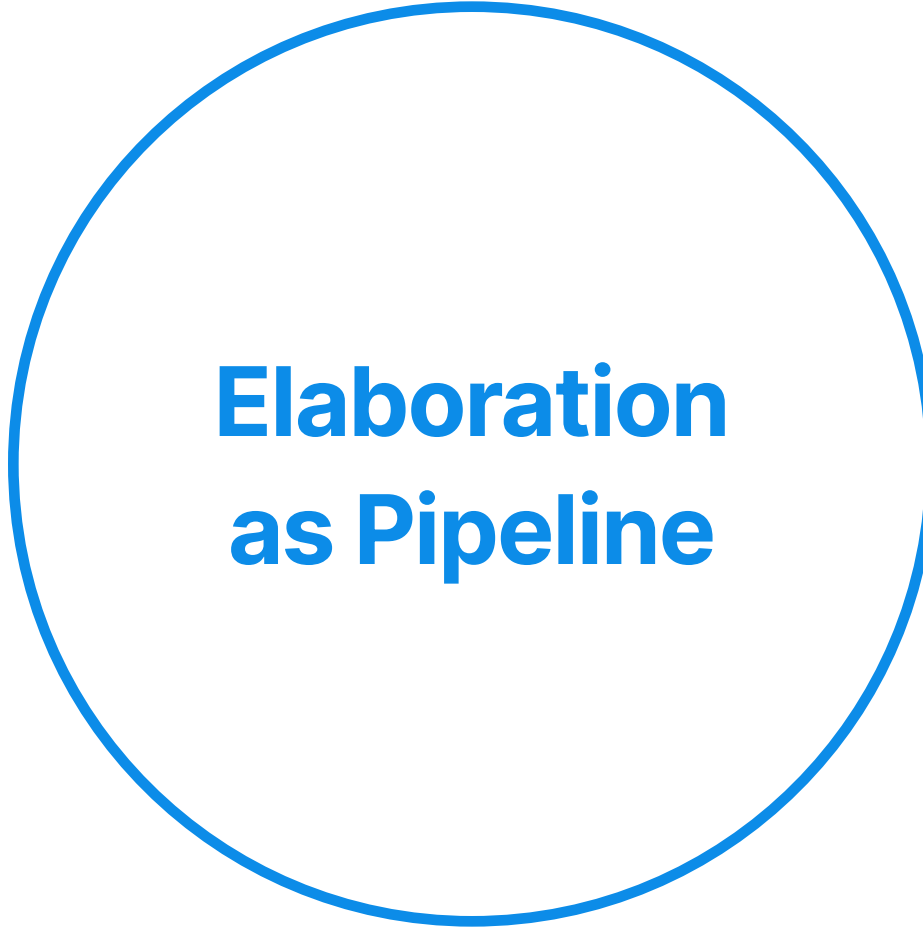
Principles



Cognitive Representations

Words and Phrases vs
Functions and
Variables

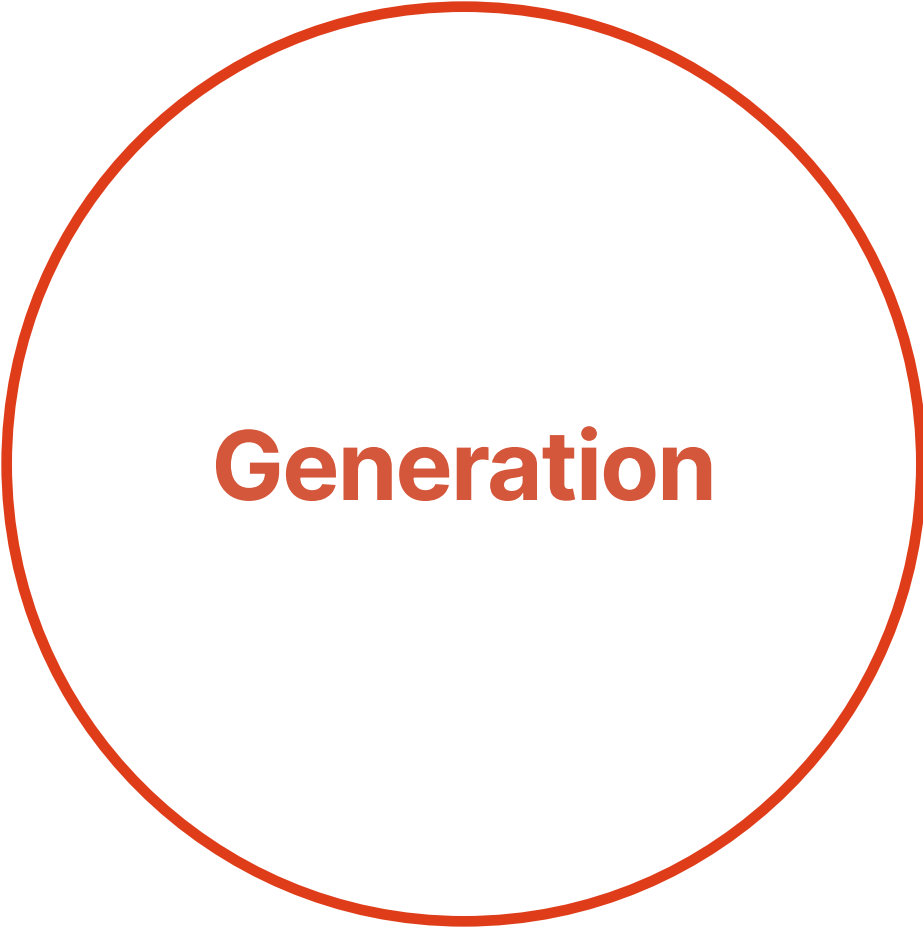
Both are based on building
blocks and specific rules to
combine them to reach
complex meanings.



Elaboration as Pipeline

Understanding text
vs Deciphering code

This requires a multiphasic
process from elaboration to
high level inferences.



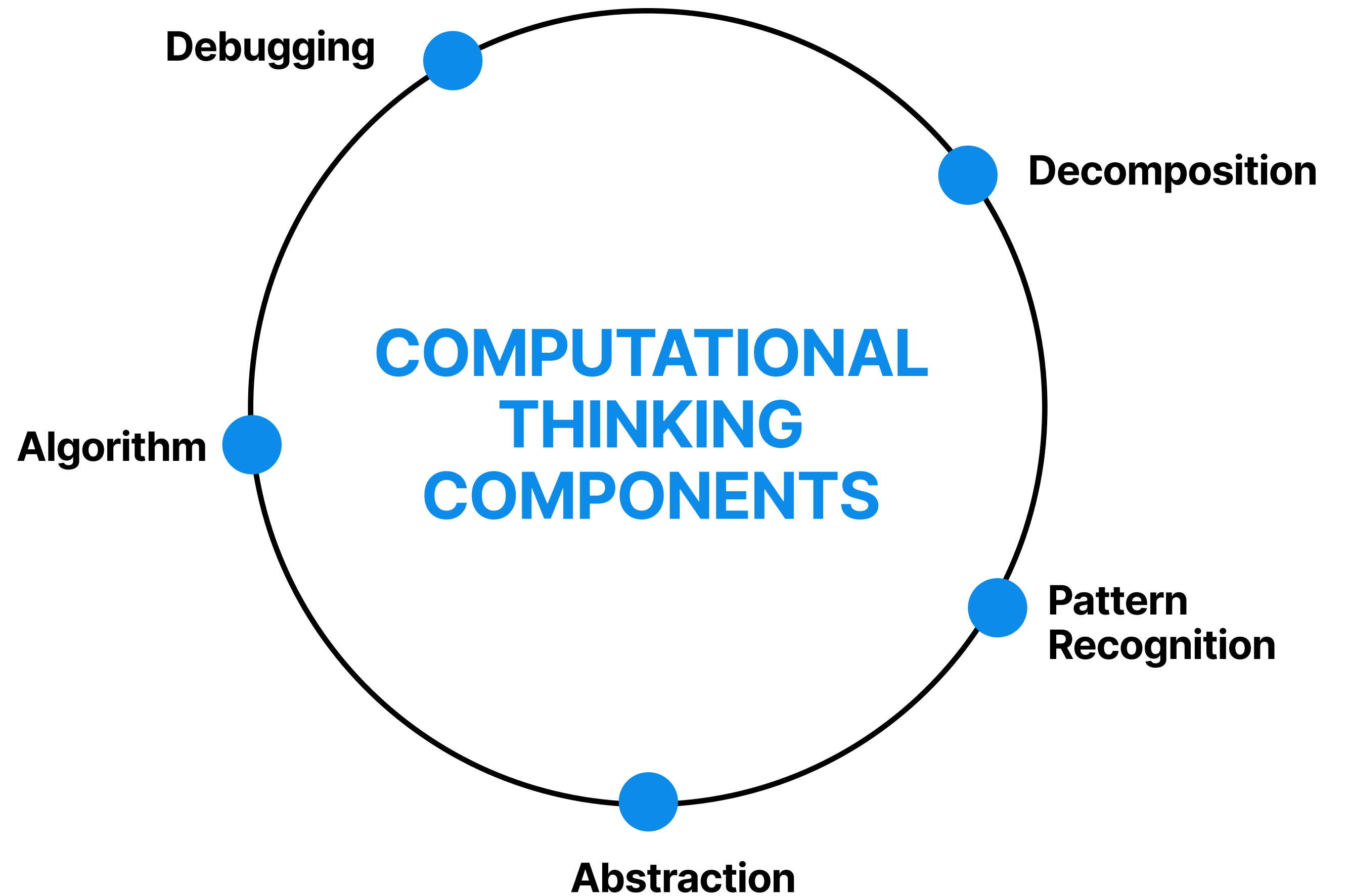
Generation

Expressing with words
vs Writing code

Need of determining, within an
analytical approach, an overall
structure and then plan a top-
down strategy for generation.

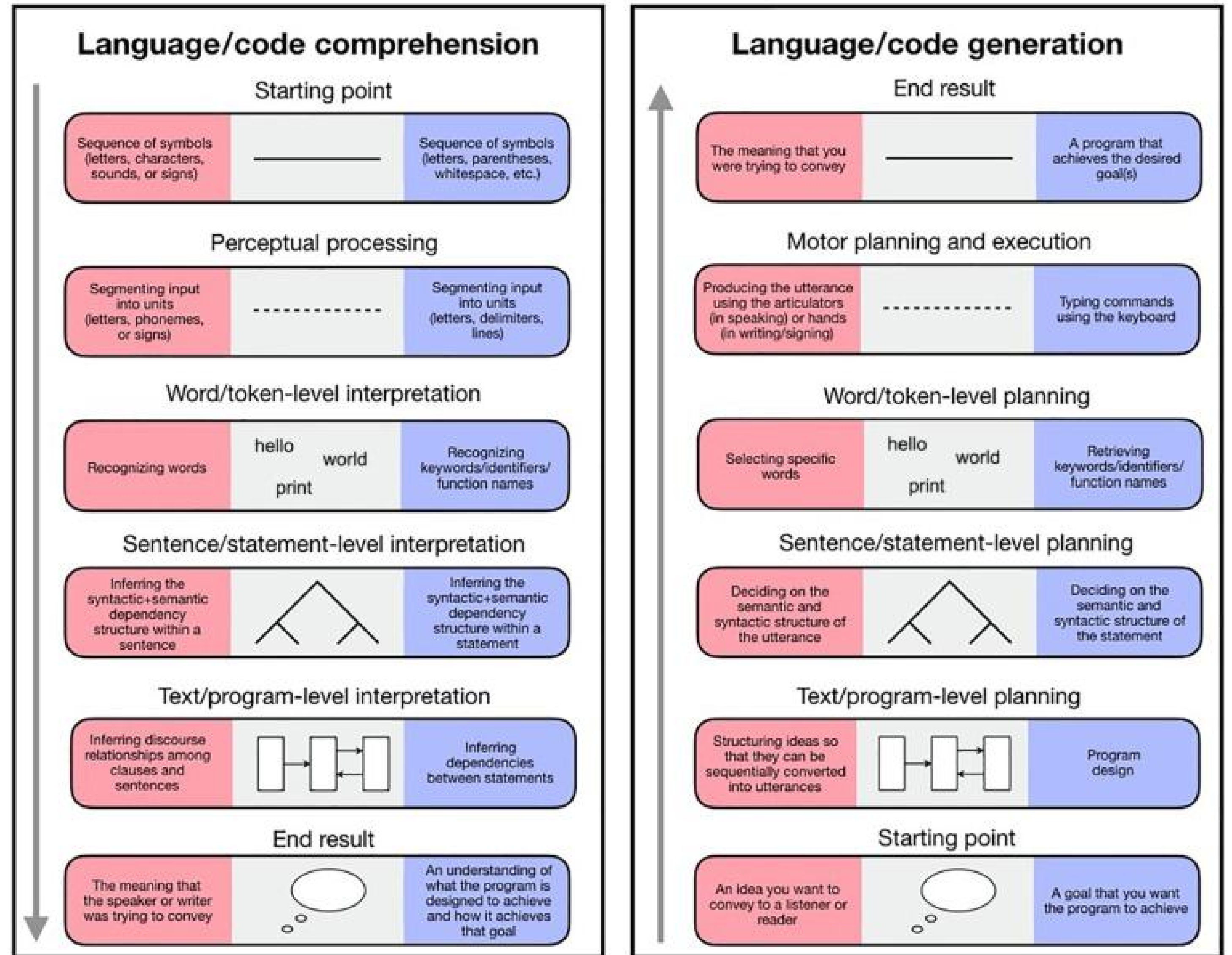
Computational Thinking

“the mental process of formulating a problem and expressing its solutions so they can be effectively implemented by an external executor both human or artificial”



The side of Linguistic Proficiency

think about **natural language programming**

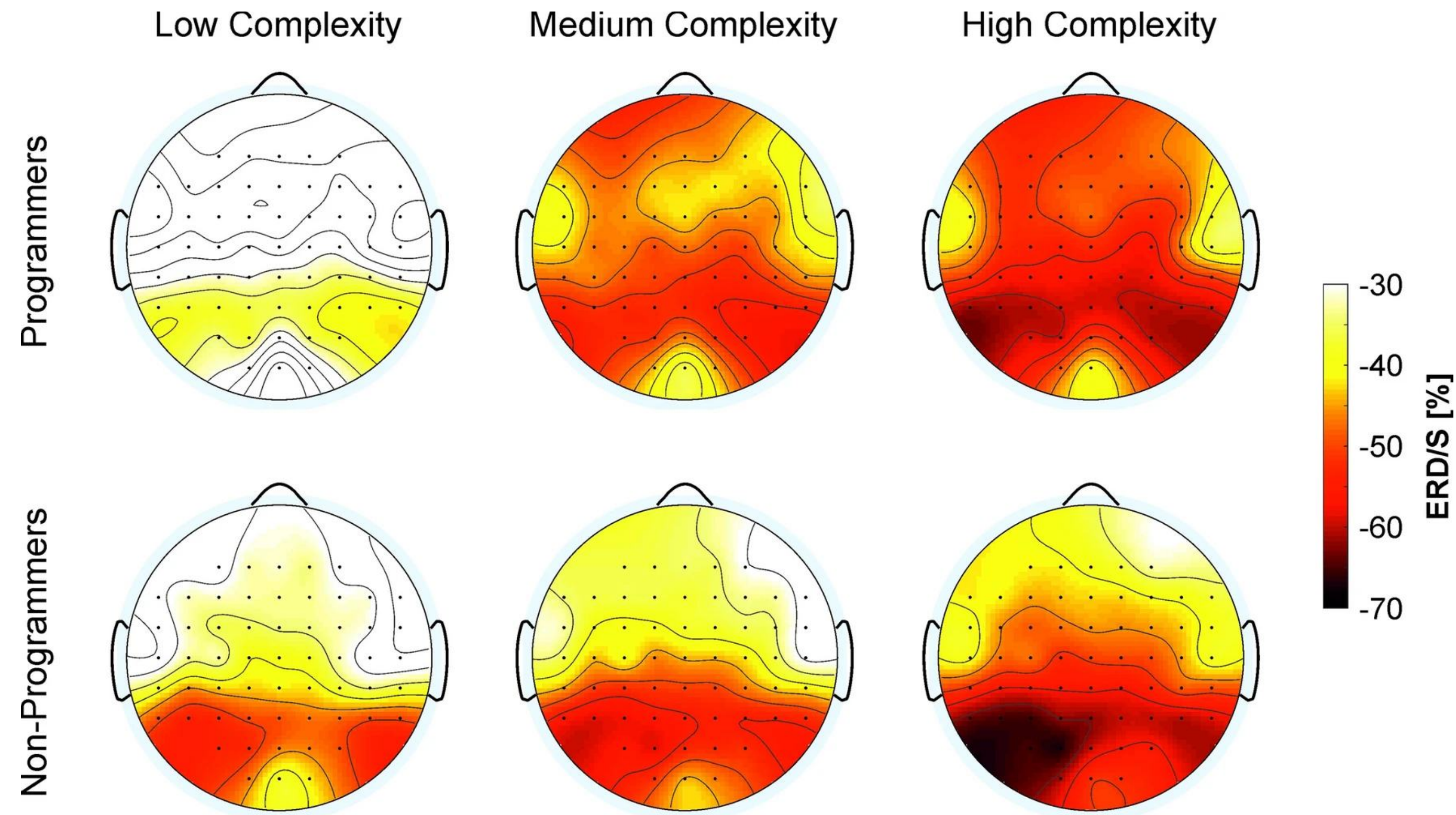


Trends in Cognitive Sciences

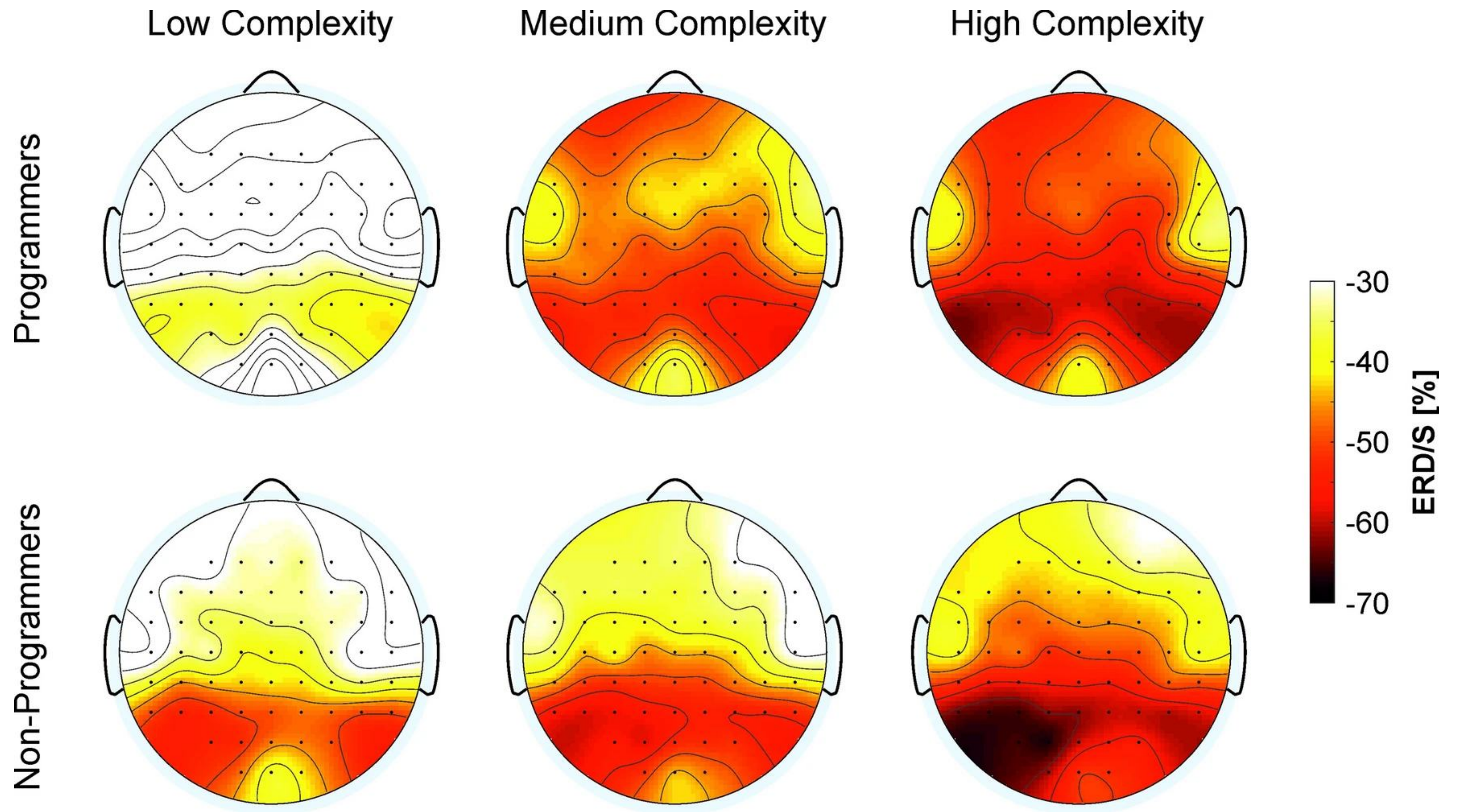
a very good dev's mind could be like a poet's one



What neurosciences say



Topographical plots of alpha ERD/S. Topoplots showing alpha ERD/S in programmers and non-programmers in the three complexity levels (low, medium, high) of the Figural Inductive Reasoning task. Only negative values are displayed (ERD). Lower values represent a more pronounced ERD (red), higher values a less pronounced ERD (yellow).



In the development lens

Literacy and numeracy rely on specific brain areas, but these did not always exist, they evolved from older phylogenetic and ontogenetic systems (**re-cycling processes**).

These abilities emerge from a combination of visual reasoning and symbolic reasoning:

- one rooted in high-level cognitive processing for problem solving,
- the other in language systems.

Why is this relevant to SFS?



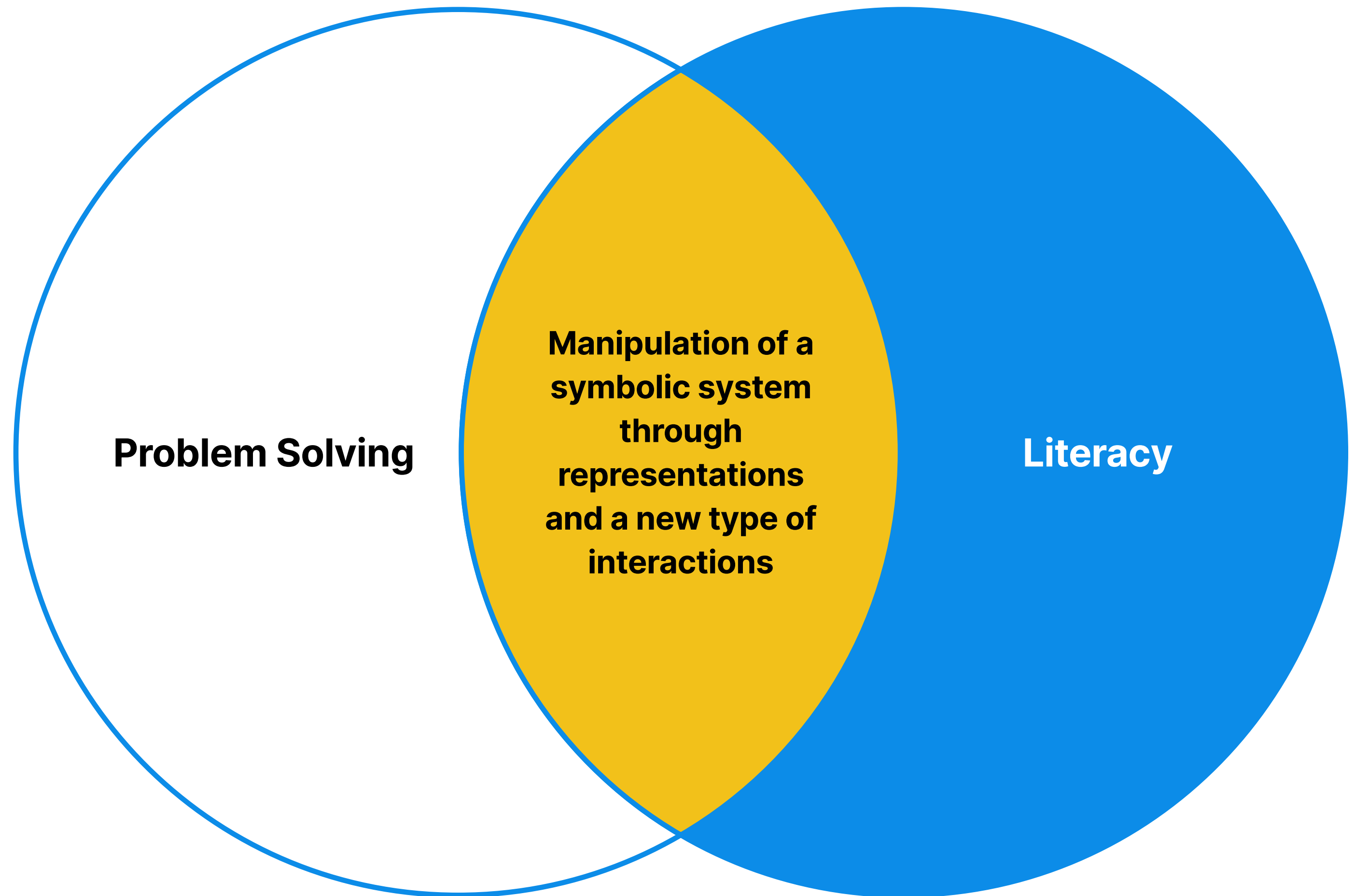
By all accounts it doesn't make any sense

Learning a symbolic system and a specific language

- implies a cultural and language link
- reflection and impact on education



**Past's education was
more about learning
math and reading,
now it's code**





**Introducing a *new literacy* is
not easy and
not without consequences**

as it empowers and affect one's own individual
approaches to elaborate information.

Thank you

Questions?

We can have a chat or (in case of low social battery)
you can find me here:

✉ giulia.paludo@unitn.it

 www.linkedin.com/in/giulia-paludo



UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA



**UNIVERSITÀ
DI TRENTO**