



---

## Breaking it Down and Putting it Back Together: Critical Thinking Skills for Teens

Kelsey Hughes, PGCMLS and Carrie Sanders, MSL



# TRANSFORMING TEEN SERVICES **TRAIN THE TRAINER**

- 
- Agreements
  - Icebreaker
  - What is Computational Thinking?
  - Some examples
  - Bringing it back to your library

# What will you learn today?

Participants will be able to:

- Articulate core concepts of computational thinking (CT)
- Begin designing activities for and with teens that connect to CT
- Understand the value of integrating CT into activities for and with teens
- Connect CT to pre-existing library activities and services.

# ROPES

**RESPECT** yourself & others, take **RISKS**,

Be **OPEN**,

**OUCHIE/OOPS** is ok

**PARTICIPATE** and **PASS**

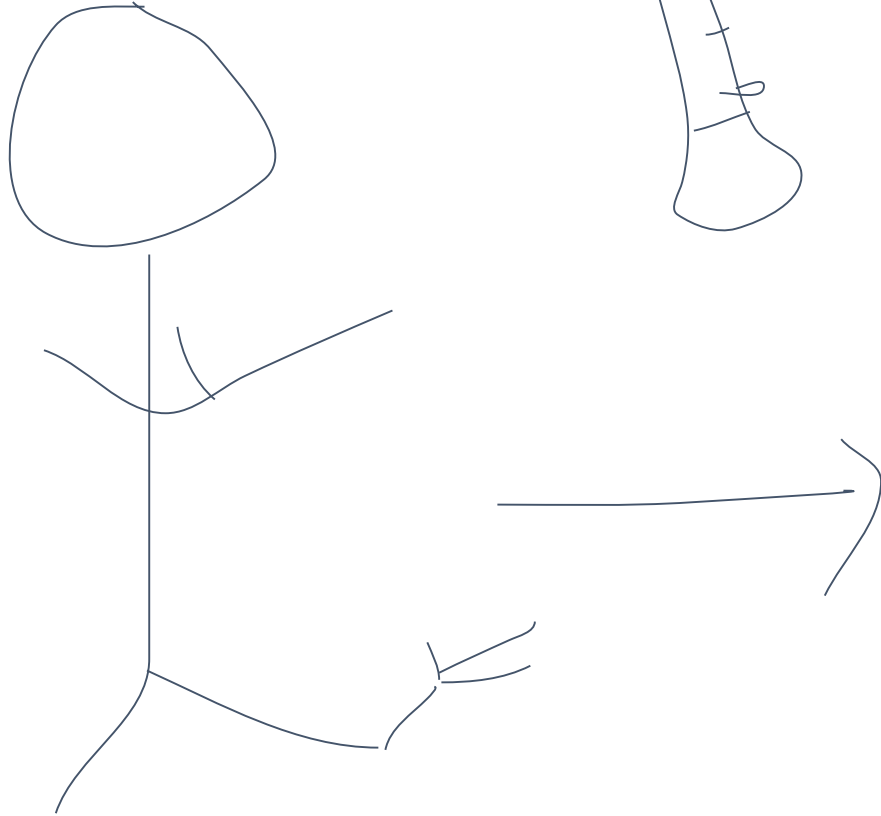
**ESCUCHAR** (listen),

be open to the **EXPERIENCE**

**SENSITIVE** and **SAFE SPACE**



**Let's play Pictionary!**



# COMPUTATIONAL THINKING

## DECOMPOSITION

Breaking big problems into smaller, easier to manage problems



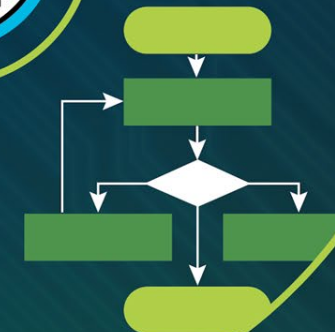
## PATTERN RECOGNITION

Analyze & look for a repeating sequence



Remove parts of a problem that are unnecessary and make one solution work for multiple problems

## ABSTRACTION



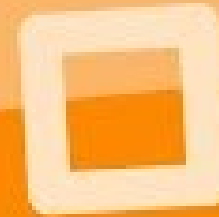
Step-by-Step instructions on how to do something

## ALGORITHM DESIGN

How does  
Pictionary use  
Computational  
Thinking?



# WHAT IS COMPUTATIONAL THINKING?







# COMPUTATIONAL THINKING

## DECOMPOSITION

Breaking big problems into smaller, easier to manage problems



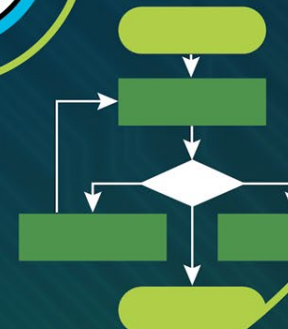
## PATTERN RECOGNITION

Analyze & look for a repeating sequence



Remove parts of a problem that are unnecessary and make one solution work for multiple problems

## ABSTRACTION



Step-by-Step instructions on how to do something

## ALGORITHM DESIGN

# The Computational Thinkers

## concepts



### Logic

Predicting & analysing



### Evaluation

Making judgements



### Algorithms

Making steps & rules



### Patterns

Spotting & using similarities



### Decomposition

Breaking down into parts



### Abstraction

Removing unnecessary detail



## approaches



### Tinkering

Changing things to see what happens



### Creating

Designing & making



### Debugging

Finding & fixing errors



### Persevering

Keeping going



### Collaborating

Working together

We're all computational thinkers here!

When you think about it, whether we're parents, pupils or teachers - we're all natural computer scientists, capable of computational thinking.

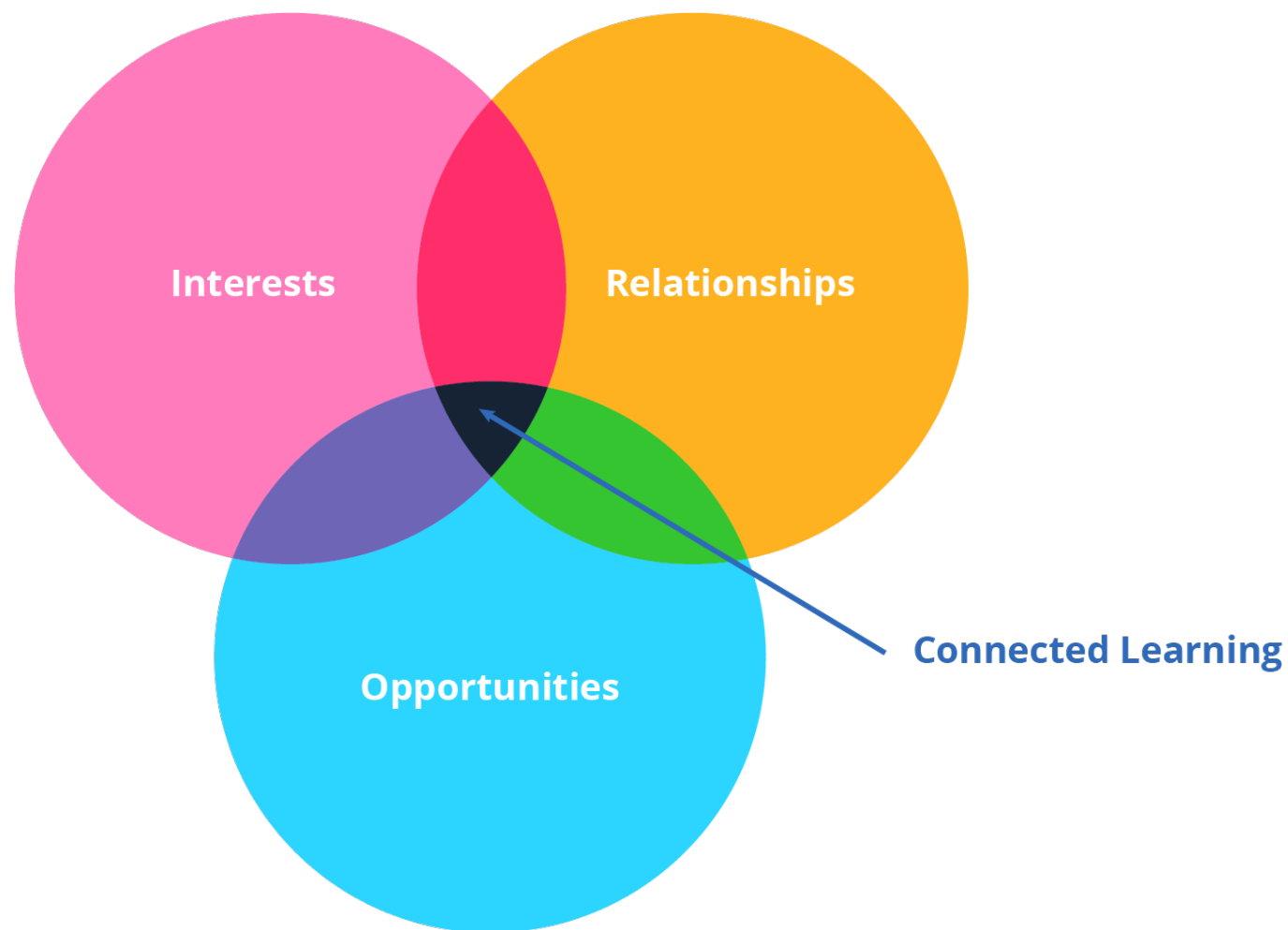
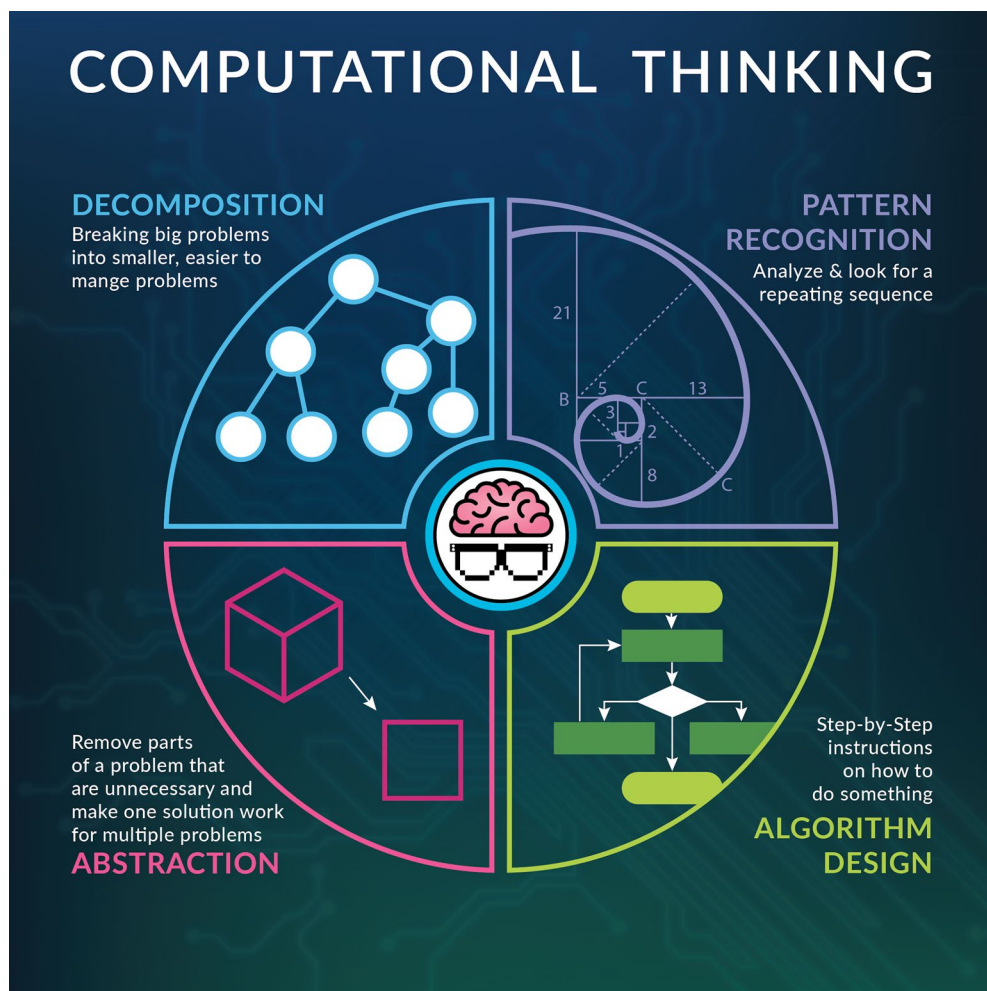
Our brains, like computers, process, debug and make simple algorithms every day!

CAS Barefoot

Supported by BT



# What does CT look like at your library?





**[forms.gle/DfDAKowJ14UH47An8](https://forms.gle/DfDAKowJ14UH47An8)**

