

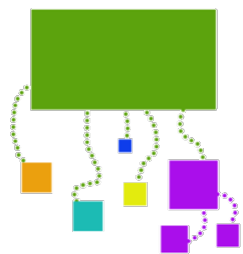
A Super Quick Overview of Computational Thinking for Young Learners by Gail Lovely

Defining CT:

There are MANY ways to define Computational Thinking... my own working definition to think with is:

Computational thinking (CT) is a creative way of thinking that encourages young children to be systematic problem-solvers who can identify problems and generate step-by-step solutions that can be communicated and followed by computers or humans.

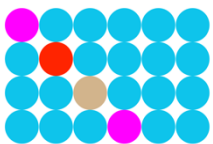
Components of Focus with Young Learners:



Decomposition:

Breaking a problem into smaller "chunks" (and then using those "chunks" to replicate, solve a problem or understand a process.)

Examples: steps in a process, parts of a story, sounds in a word



Pattern Recognition:

Making connections between similar problems and experience and/or finding patterns and extending and testing them. Requires noticing and describing attributes and characteristics.

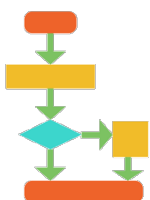
Examples: sequences of events (lunch before recess), robots need power to "work", word order



Abstraction:

Filtering out (ignoring) the characteristics we don't need (or focusing on only the characteristics which "matter") AND Identifying important information while ignoring unrelated or irrelevant details.

Example: when sorting something by color, the shape doesn't matter



Algorithms:

Creating and testing step-by-step plans to solve problems or achieve results.

Examples: Solving disputes between friends, building things, stacking blocks

