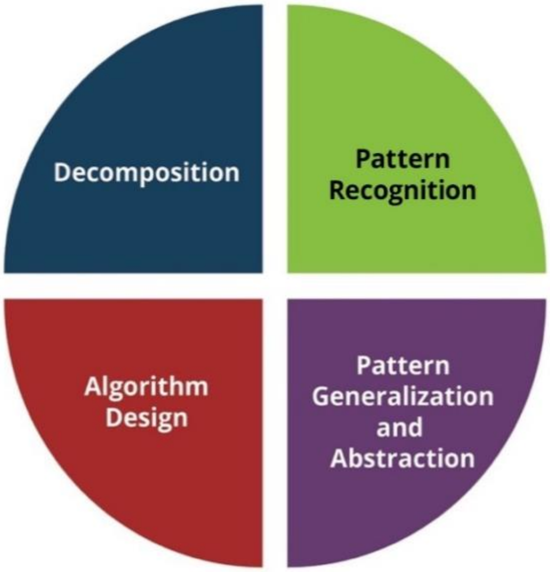
Computational Thinking (CT)

TEA Texas Education Agency

Computational Thinking (CT) is a thinking process used to solve problems logically and methodically. Computational Thinking promotes a purposeful, describable, and replicable method of problem solving.



Steps in Computational Thinking
Decomposition
Breaking a problem into smaller pieces, or subprocesses, to describe, understand, or better solve the problem.
Pattern Recognition
Identifying similarities or common differences that help us describe patterns and make predictions.
Pattern Generalization and Abstraction
Organizing relevant information and recognizing commonalities between patterns to make one solution work for multiple problems.
Algorithm
A process, or list of steps, followed to solve a problem or complete a task.

Decomposition

The computational thinking process begins with the **decomposition** of the problem—breaking down the process into a set of smaller subprocesses that allows students to describe, understand, or better execute the process. Decomposition includes dividing a task into a sequence of subtasks or identifying elements or parts of a complex system. For example:

- Providing directions to a location.
- In mathematics, numbers are decomposed in a variety of ways.
  - 247 is 200 + 40 + 7
  - 247 is 2(100) + 4(10) + 7
  - 247.82 is 2 * 102 + 4 * 101 + 7 * 100 + 8 * 10 -1 + 2 * 10 -2

Pattern Recognition

Once a problem is decomposed, similarities or common differences are identified that will help in making predictions or discovering shortcuts. This step is called **pattern recognition**. Examples include:

- Shoppers looking for the shortest line at the supermarket when choosing a register for check out or when watching the stock market to decide when to buy and sell stocks. Drivers also look for patterns when driving in traffic, helping them decide if and when to change lanes. Scientists look for patterns in data to derive theories and models.

Pattern Generalization and Abstraction

After identifying patterns, the next step is to **abstract** and **generalize** the patterns. In this process, the relevant information for the problem being solved is kept and irrelevant information is dismissed.

- **Abstraction** is used to organize or categorize things (a human is a mammal, a mammal is an animal, and so on). Abstraction reveals the "big picture", so reasoning can occur without thinking about details. Abstraction allows learners to transfer learning or learn by creating an analogy. For example:
  - A summary is given when telling a story or describing a movie to a friend; not every detail is provided.
  - A world map is an abstraction of Earth in terms of longitude and latitude, helping to describe the location and geography of a specific place.
  - A book report is an abstraction because it is a summary only covering the theme or key aspects of the book.
- When learners **generalize** patterns, they identify common or shared characteristics between two domains or problems so that models or solutions of one can be adapted or applied to the other. When studying trends, norms, outliers, and scalability, patterns are used to generalize. For example:
  - When studying mammals, learners know they are warm-blooded, give live birth, have hair, etc. An elephant is a mammal. Therefore, the elephant is warm-blooded, gives live birth, has hair, etc.
  - Pattern generalization is utilized when using a search engine. Pattern generalization identifies and suggests popular keywords, such as through auto-complete or autocorrect. Online shopping companies use pattern generalization to predict what will be of interest to their customers and make recommendations accordingly.

Algorithm Design

The final step is to **design an algorithm** or develop a step-by-step strategy for solving a problem. Algorithmic thinking involves both the creation and the execution of an algorithm. For example:

- When a grandmother writes down a family recipe for her signature dish, an algorithm is created that others can follow to replicate the dish.
- When using online maps to get directions to a friend’s house, there is a specific sequence of steps to follow.
- A science lab experiment has a set of instructions to carry out to collect and analyze data.

Once an algorithm is designed, it is essential to evaluate the solution for correctness. The algorithm should meet all the requirements to solve the problem and the algorithm should perform with efficiency and reliability. Optimizing algorithms is critical because solutions should be usable, efficient, reliable, and effective. If a solution can be solved with 9 steps instead of 14, the algorithm should be optimized for efficiency. When folding a paper airplane, a test is completed to check its ability to fly. Then modifications or revisions might be made to the design to improve the outcome. All of these demonstrate optimization and evaluation of a solution.