

# Abstraction

## **The four cornerstones of computational thinking**

1. **Abstraction** = gathering of general relevant detail and the filtering out/ignoring of the unnecessary characteristics of a problem.
2. **Decomposition** = Looking for patterns among and within smaller problems that make up the more complex problem.
3. **Algorithms** = A well-defined procedure of instructions and steps needed to carry out a task so a computer can solve a problem.
4. **Pattern recognition** = Finding the similarities or patterns among small, decomposed problems that can help us solve more complex problems more efficiently.

## **Abstraction**

The level of complexity and detail required is decided and more complex details are suppressed below the current level on which a person interacts with the system.

There are many levels of abstraction depending on the level required. Complex processes are hidden from the user to make it look simple. It will not remove detail but remove unnecessary detail from the user's experience.

## **Why is abstraction important?**

Abstraction allows us to create clear models – a general idea of what the problem is and how to solve it. Abstraction instructs us to remove all specific detail and any patterns that will not help us solve the problem.

However, higher level of complexity are needed in abstraction to make things look more realistic.