Computational Thinking Concepts Guide

Overview

In this guide you will find eleven terms and definitions for Computational Thinking (CT) concepts. These concepts can be incorporated into existing lesson plans, projects, and demonstrations in order to infuse CT into any disciplinary subject.

Concepts, Definitions, and Teaching Tips

Abstraction is identifying and extracting relevant information to define main idea(s)

Teaching Tips:
- Show how a daily planner uses abstraction to represent a week in terms of days and hours, helping us to organize our time
- Find the main idea (thesis) of a short story or article to help students understand the important information

Algorithm Design is creating an ordered series of instructions for solving similar problems or for doing a task

Teaching Tips:
- Give the example of how in mathematics, when we add and subtract fractions with different denominators, we follow an algorithm
- Give the example of when a chef writes a recipe for a dish, she is creating an algorithm that others can follow to replicate the dish

Automation is having computers or machines do repetitive tasks

Teaching Tips:
- Show how automation can be used to perform tasks that would take a very long time to complete using a manual process, such as identifying the migration patterns of a specific demographic based on census data

Data Collection is gathering information

Teaching Tips:
- Have students engage in the process of collecting data by having them gather the birthday and gender of their peers and record it in a spreadsheet

Data Analysis is making sense of data by finding patterns or developing insights

Teaching Tips:
- Ask students to formulate their own questions that can be addressed with data collection; work through data together to answer them
- Using data sets with anonymized personal data, such as height, shoe size, favorite color, etc., show how data analysis can be used to highlight information that is meaningful and relevant to your students

Data Representation is depicting and organizing data in appropriate graphs, charts, words, or images

Teaching Tips:
- Plot data manually on the whiteboard or via projector so that your students can see the process of how the organization unfolds
- Ask your students which visual representation is best for a given data set and work through some less relevant visualizations first. Discuss as a class why the data should be represented in one way or another.

Decomposition is breaking down data, processes, or problems into smaller, manageable parts

Teaching Tips:

- In mathematics, we can decompose a number such as 256.37 as follows: $2*10^2+5*10^1+6*10^0+3*10^{-1}+7*10^{-2}$
- In science we decompose a projectile's velocity into its components along the x- and y-axis

Parallelization is simultaneous processing of smaller tasks from a larger task to more efficiently reach a common goal

Teaching Tips:

- An example of parallelization in computing is when a single task (such as the analysis of a DNA sequence) is broken into smaller tasks and simultaneously analyzed by different computers so that the analysis can be processed more efficiently
- Define the common goal at the outset, and then have your students work in groups. Assign different tasks to each group to work toward the common goal.

Pattern Generalization is creating models, rules, principles, or theories of observed patterns to test predicted outcomes

Teaching Tips:

- In mathematics, we write generalized formulas in terms of variables instead of numbers so that we can use them to solve problems involving different values
  - The slope of any straight line can be described as a function of y = mx + b
- In science, we use theories to describe the generalized mechanism by which natural phenomena occur

Pattern Recognition is observing patterns, trends, and regularities in data

Teaching Tips:

- Have your students identify trends in stock price cycles that may suggest when they should be bought and sold

Simulation is developing a model to imitate real-world processes

Teaching Tips:

- Have your students illustrate the movement of a solar system by modeling the gravitationally curved path of an object around a point in space

---

More reference docs, lesson plans, and demonstrations can be found on Google's Exploring Computational Thinking website (g.co/exploringCT)