

Algorithms & Programming: Modularity (1) Grade: 5

Standard: 5.AP.M.01

Decompose a large problem into smaller, manageable sub-problems and then further into sets of sequenced instructions to facilitate the program development process.

Essential Skills

Devise **algorithms** to solve identified sub-problems

Demonstrate how combinations to the solutions of sub-problems can simplify writing programs to solve complex problems.

Essential Questions

What is the advantage of **decomposing** problems?

How does decomposing a problem into sub-problems help when writing a complex **computer program**?

Explanation

Students will be able to break down, or **decompose**, complex problems into smaller problems that are easier to solve and manage. They can then express the solutions to those smaller problems as a series of instructions or an **algorithm**. Students should understand how to combine the algorithms addressing the smaller problems in order to solve the more complex problem, and how that process facilitates program development. Additionally, students should demonstrate an understanding that modifying solutions or finding errors in instructions is easier when addressing smaller subproblems than large complex problems and that decomposition enables different people to work on different parts of the same project simultaneously.

Think of this as similar to....

When you make a holiday meal, you may start by preparing the main course. Someone else may set the table. Yet another person makes the dessert. You combine the parts to create the whole meal.

Implementation Examples—What would this look like in the classroom?

Title	Description	Link	Content Connection & Notes
Abstraction Unplugged	<p>Grade 4--Students will be presented with a Scratch project that they will decompose with their partners without having access to its code and without access to a computer. They will propose the smaller chunks of programming that would be most helpful in programming the project.</p> <p>Grade 5--Students propose an arrangement to assemble the smaller chunks of programming in order to create the whole Scratch program. Write the algorithms the sub programs (smaller chunks) in Scratch and determine if they work as desired.</p>	Abstraction Unplugged	
Functions in Minecraft	<p>Grade 4--Students find reusable patterns in code when they name that chunk of code they have created a function. That function can accomplish a portion of what the puzzle asks them to do.</p> <p>Grade 5--Students create functions and use them repeatedly in their program to tell the computer to run that chunk of code in different environments. They should be able to describe their solution to the larger puzzle in terms of the functions they have created. (This would be done in the Free Play portion of the lesson.)</p>	Functions in Minecraft	
Energy	<p>Grade 5--In order to plan a model to demonstrate that energy in animals' food was once energy from the sun, the students will decompose the problem into how animals get energy (from food) what is the food (other animals and ultimately plants) and where plants get energy (from the sun). Students then plan their model and plan how their program (in Scratch or another programming language) will demonstrate the different transfers of energy. Once the plan is complete, students create each part of program and adjust each so the parts work together.</p>	Energy	This lesson also aligns with NGSS 5-PS3-1

Standard: AP.M.01 Grade:5

These annotations are a collaboration between [Maryland Center for Computing Education](#) and the [Maryland State Department of Education](#).