

# Algorithms & Programming: Modularity (1)      Grade: 3

Standard: 3.AP.M.01

**Decompose** a simple problem into a precise set of sequenced instructions.

Essential Skills

Devise an **algorithm**, a set of ordered instructions, to solve a problem.

Essential Questions

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

Explanation

Students will be able to break down, or **decompose**, a simple problem and express the solution as a series of instructions or an **algorithm**.

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
<b>Determine Area by Decomposition</b>	<b>Grade 3</b> -- This a sequence of three lessons. Day 1 covers defining the computer science vocabulary term "decomposition", defining area, and relating the two. Day 2 covers decomposing a rectilinear figure into cubic squares. In Day 3, students apply their learning to determine the area of letters represented on a grid.	<a href="#">Determine Area by Decomposition</a>	This lesson also aligns with <b>Math</b> 3.MD.C.5 and 3.MD.C.6
<b>Nested Loops in Maze</b>	<b>Grade 3</b> --Students will use the puzzles in this CS Fundamentals lesson to decompose (break down) a long code into smaller sections of code. As they play with the Bee and Plants vs Zombies students further decompose the code into smaller sections and examine them for patterns.	<a href="#">Nested Loops in Maze</a>	
<b>Measuring Jack's Path</b>	<b>Grade 3</b> --Students mark the start and stop point of Jack (a Code and Go mouse or other floor robot) when it moves forward by pushing the forward arrow one time. Students measure this distance to the nearest quarter inch. Using this measurement, students create a maze Jack can travel through. Once maze is complete, students create a code for Jack to complete each section of the maze to get to the end. Students test their code for each section of the maze, and have a peer test the code to ensure it works well. Students add on to their code as the robots successfully navigates each section of the maze. Students then measure the distance Jack travelled from start to end of the maze while executing all the code. The class comes together and records the distance Jack traveled for their particular maze (nearest quarter inch), and records data on a line plot.		This lesson also aligns with <b>Math</b> 3.MD.B.4; floor robots are used.

Standard: AP.M.01 Grade:3:

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