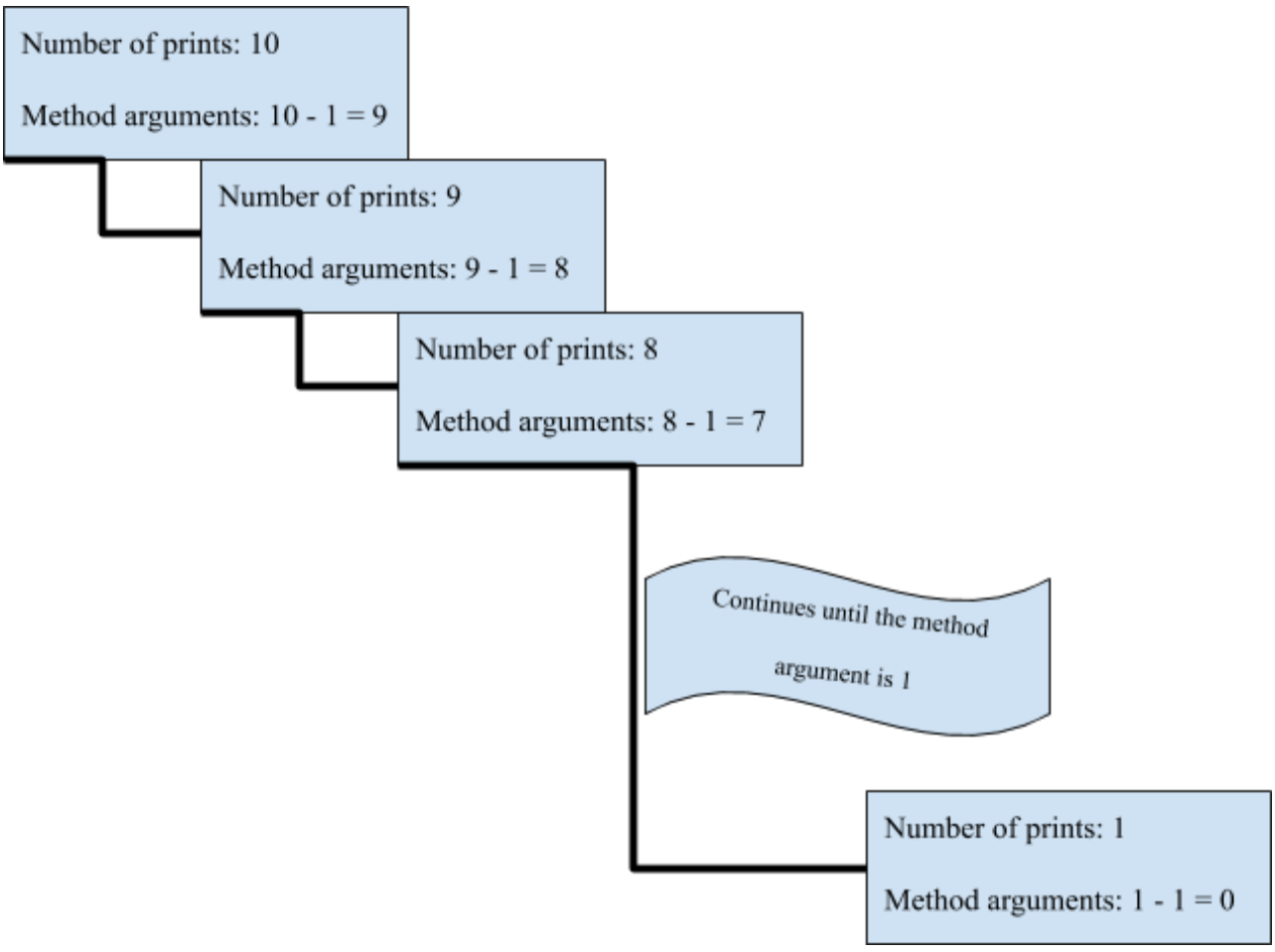
Chapter 16 - Recursion

Chapter 16 introduces the idea of recursion, which in computer science is a problem-solving method that involves a method that calls itself until a specific condition is met. The basic idea is to break down the problem into smaller and smaller problems that are identical to the overall problem. This chapter also covers the differences between the base case and the recursive call in order to avoid calling the method infinitely.

In Java, methods are used as a tool to solve problems using recursion. Similar to loops, recursion repeats a segment of code multiple times until a particular condition is met. The difference is that in recursion, the scale of the problem gets successively smaller after each iteration. ProgramOneCSixteen.java implements a recursion on a simple problem in order to explain the idea of "breaking down the problem into smaller pieces".

The following diagram represents the use of recursion in ProgramOneCSixteen.java:



*Task: Print the number ten a total of ten times in a line. After printing the number ten times, start a new line and print the number nine a total of nine times. Then, start another line and print the number eight a total of eight times. This should repeat until the number is one and is printed only once.

There are two critical aspects inside of a recursion method, the base case and the recursive call. The recursive call is the segment of code in the method that allows the method to call itself. The base case is a parameter for the recursive call, similar to a control variable in a for loop. The base case is essential in almost all recursion methods because, without a parameter, the recursive call would either keep calling itself an infinite number of times or cause a run-time error because of invalid arguments. ProgramTwoCSixteen.java provides an example of a base case and a recursive call, outlying the importance of a base case.