**Recursive Solutions**

**What Is Recursion?**

* Recursion is an extremely powerful problem-solving technique.
* Problems that at first appear to be quite difficult often have simple recursive solutions.

**Recursion breaks a problem into several smaller problems**.

* Recursion is a powerful problem-solving technique that uses a ***divide-and-conquer*** approach.
* You break a problem down into simpler (usually smaller) versions of itself that are the ***exact same type*** as the original problem.
* That is, a recursive solution solves a problem by solving a smaller instance of the same problem!
* It then solves this new problem by solving an even smaller instance of the same problem.
* Eventually, the new problem will be so small that its solution will be either obvious or known when it reaches a ***base case***.

**Recursive Solutions have the following the following form:**

1. A recursive function calls itself
2. Each recursive call solves an identical, but smaller, problem
3. Test for base case enables recursive calls to stop
4. Eventually, one of the smaller problems must be the base case

**As you attempt to construct a new recursive solution, there are four questions to keep in mind:**

1. How can you define the problem in terms of a smaller problem of the same type?
2. How does each recursive call diminish the size of the problem?
3. What instance of the problem can serve as the base case?
4. As the problem size diminishes, will you reach this base case?

**Example:**

* For example, suppose that you could solve problem *P*1 if you had the solution to problem *P*2, which is a smaller instance of *P*1.
* Suppose further that you could solve problem *P*2 if you had the solution to problem *P*3, which is a smaller instance of *P*2.
* If you knew the solution to *P*3 because it was small enough to be trivial, you would be able to solve *P*2. You could then use the solution to *P*2 to solve the original problem *P*1.

**Iteration vs. Recursion**

* Recursion is an alternative to **iteration**.
* An iterative solution involves loops, a recursive solution involves a function’s repeated calls to itself.

**Not all recursive solutions are better than iterative solutions**

* In fact, some recursive solutions are impractical because they are so inefficient.
* Recursion, however, can provide elegantly simple solutions to problems of great complexity.

**Example:**

* As an illustration of the elements in a recursive solution vs. an iterative solution, consider the problem of looking up a word in a dictionary.
* Suppose you wanted to look up the word “vademecum.”
* Imagine starting at the beginning of the dictionary and looking at (iterating over) every word in order until you found “vademecum.”
* That is precisely what a ***sequential search*** does, and for obvious reasons, you want a faster way to perform the search.