**Recursion versus Iteration**

* A recursive approach mirrors the problem that we are trying to solve.
* A recursive approach makes it simpler to solve a problem, which may not have the most obvious answers.
* But, recursion adds overhead for each recursive call (needs space on the stack frame)

**Recursion**

* Terminates when a base case is reached.
* Each recursive call requires extra space on the stack frame (memory).
* If we have a infinite recursion, the program may run out of memory and gives back stack overflow.
* Solutions to some problem are easier to formulate recursively.

**Iteration**

* Terminates when a condition is proven to be false.
* Each iteration does not require any extra space.
* An infinite loop could loop forever since there is no extra memory being created.
* Iterative solutions to a problem may not always be as obvious as a recursive solution.

**Note:**

* Recursive algorithm have two types of cases, recursive cases and base cases.
* Every recursive function case must terminate at base case.
* Generally iterative solutions are more efficient than recursive solutions.
* A recursive algorithm cab be implemented without recursive function call using a stack, but it’s usually more trouble than its worth. That means any problem that can be solved recursively can also be solved iteratively.
* For some problems, there are no obvious iterative algorithms.
* Some problems are best suited for recursive solutions while others are not.