Recursion

Recursion is when a function is calling itself (multiple time) with the smaller input until it reaches to base case/ trivial case.

Thus, resulting in reducing the size of problem (i.e., Converging the problem towards base case/ trivial case).

Math behind the Recursion

**PMI (Principle of Mathematical Induction)**

For a function f(x)

**Step 1:** To **Prove** for lower value like f(0) or f(1) is true **| [ Base Case/ Trivial Case ]**

**Step 2**: **Assume** f(k) is true **| [ Induction hypothesis]**

**Step 3: Prove** f(k+1) is true using step 1 and step 2 **|** **[ Induction Step ]**

**Extended form of PMI**

Step 1: Assume f(0), f(1), f(2), … f(k) is true

Step 2: Using all f(0)…f(k), prove f(k+1) is true

**Note:** Fibonacci series requires 2 base cases in a recursion. (**Extended PMI**)

RecursionError: maximum recursion depth exceeded in comparison

The reason we get it is because every recursion call uses some system memory and python has certain memory limit to use by recursive call if it gets exceeded, it gives us a recursion error.

import sys

sys.setrecursionlimit(3000) # to increase the recursion limit in python