RECURSION

**Step1) What is the simplest possible input? Our base case.**

ex – sum of all non-negative integers – sum(0) -> 0

**Step2) Play around with examples and visualize!**

ex – n = 1 (1)

n = 2 (1+2)

n = 3 (1+2+3)

n = 4 (1+2+3+4)

**Step3) Relate hard case to simpler cases.**

ex – Can you relate sum(4) and sum(5)

Can you relate sum(3) and sum(4)

**Step4) Generalize the pattern**

Sum(n) = {0 if n = 0}

{sum(n-1) +n}

* ex –
  + sum (5)
  + 5 + sum (5-1)
  + 4 + sum (4-1)
  + 3 + sum (3-1)
  + 2 + sum (2-1)
  + 1 + sum (1-1)
  + 0

**RECURSIVE LEAP OF FAITH**

- Assume simpler cases work out.

Q) Write a function that takes two inputs n and m and outputs the number of unique paths from the top left corner to bottom right corner of a n x m grid.

Constraints: you can only move down or right 1 unit at a time.

**Step1) What’s the simplest possible input?**

Grid\_paths(1,1) -> 1

Grid\_paths(2,1) -> 1

Grid\_paths(1,2) -> 1

Grid\_paths(1,m) -> 1

Grid\_paths(n,1) -> 1

Grid\_paths(n, m) -> 1 if n = 1or m= 1