How do we solve problems recursively?

Recursive Solutions: Fibonacci Numbers.

- Find the nth Fibonacci number
  - Each number is the sum of the previous 2 numbers, the first 2 are both 1)

1. Identify the base case:

   What is it?: When `n = 0` or `n = 1`, because there is no calculation to be made.

   What can you return?: `fib(0)` or `fib(1) = 1`

2. How to shrink?: `fib(n)` relies on `fib(n-1)` and `fib(n-2)`

3. How to get answer from smaller version?: This is essentially the definition of the fibonacci series: `fib(n) = fib(n-1) + fib(n-2)`

How do we solve problems recursively?

Code:

```
def fib(n):
   if (n < 2):
      return 1
   else:
      return fib(n-1) + fib(n -2)
```