A white paper with black text

Description automatically generated

1. The result of fib(6) will be 8. The function finds fib(8) with a process of recursive calls. Since the Fibonacci sequence is 0, 1, 1, 2, 3, 5, 8 where the nth term can be described as (n-1) + (n-2). Hence the return statement for the else statement. This uses something known as a call stack. The function call stack uses return statements from the base case (either 1 or 0), saves it, and uses it in the one before it. Since the call stack is a last in is the first out type of data structure, it removes the call from the stack and used it in the last one until it gets to fib(6).
2. The if(n<=1) is the base case since that is when recursive calls stop when the if statement runs. Since 0 and 1 are the base cases out of all positive integer values (with zero) since the first two numbers of the Fibonacci sequence is 0 and 1. So if n is either 0 or 1, it will return it.
3. The general case is the else statement after the if. In other words, the general case is when n > 1. This is where it runs the (n-1) + (n-2) part of the recursive function. As you want to make the numbers as a smaller version of themselves with the base case, where you just add 1’s and 0’s until the get the nth number of the sequence.

A diagram of mathematical equations

Description automatically generated

1. The result would just be 1. Since this is the base case.