Exercises, Week 2, Day 2

1. The fibonacci number for the values 0 and 1 is set as 1. Subsequent values are calculated as `fn(x) = fn(x-1) + fn(x-2)`. For example `fib(10) = fib(9) + fib(8)`. Write a recursive function to calculate the fibonacci number for any positive integer.

2. The previous function is inefficient as each call of the function leads to calling the function again a lot of times. How many?

3. Work out a way of caching results (using a dictionary) so that the fibonacci function is a lot more efficient.

4. Write a function that prints out whatever arguments you pass to it. That is, `myFunc (1, 'a', True)` will print out `1, a, True`.

5. Modify this so that it prints out the named arguments as well. That is, `myFunc (1, file='a', limit=True)` will print out `1, file=a, limit=True`.

6. Write a function that accepts a string, a function and following arguments and returns the result of the function applied to the string with those arguments. For example, `callStrFunc (str, find, 1, 10)` will return the result of `find (str, 1, 10)`.