

Practical 3 – Recursion

Fibonacci and Tower of Hanoi

**timing results and excel graph sheets can be found in the practical-Resources folder in sub-folder wk3.*

Warm-up questions

1. What are the two principal characteristics of a recursive algorithm?
 - **A recursive algorithm must have a base case and must change its state and move toward the base case.**
 - **A recursive algorithm must call itself recursively.**
2. **Recursion is..** **theoretically powerful and often used in algorithms that could benefit from recursive methods**
3. **True** or false: All recursive functions can be implemented iteratively.
4. True or **false**: if a recursive algorithm does NOT have a base case, the compiler will detect this and throw a compile error?
5. True or **false**: a recursive function must have a void return type.
6. True or **False**: Recursive calls are usually contained within a loop.
7. **True** or False: Infinite recursion can occur when a recursive algorithm does not contain a base case.

8. Which of these statements is true about the following code?

```
int mystery(int n)
{
    if (n>0) return n + mystery(n-1);
    return 0;
}
```

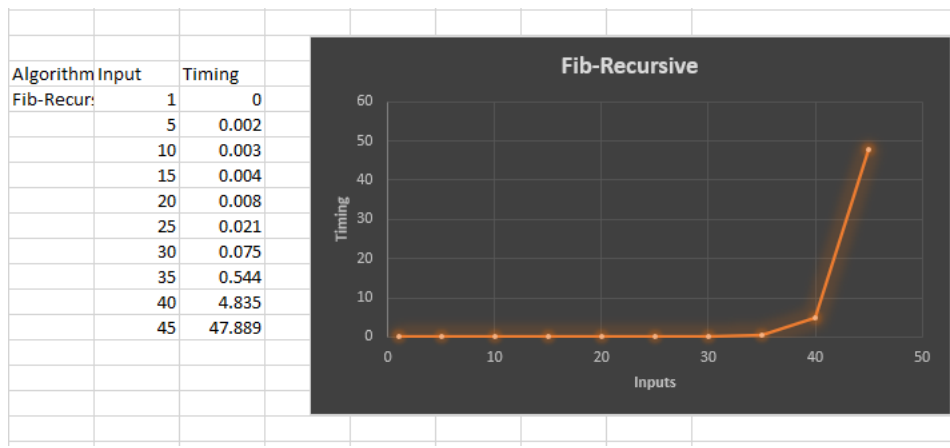
Your answer	
	The base case for this recursive method is an argument with any value which is greater than zero.
	The base case for this recursive function is an argument with the value zero.
Answer	<u>There is no base case.</u>

9. List common bugs associated with recursion?

	No base case
	Function doesn't change its state recursively to reach the base case.
	Does not call recursive method properly
	Incorrect base cases

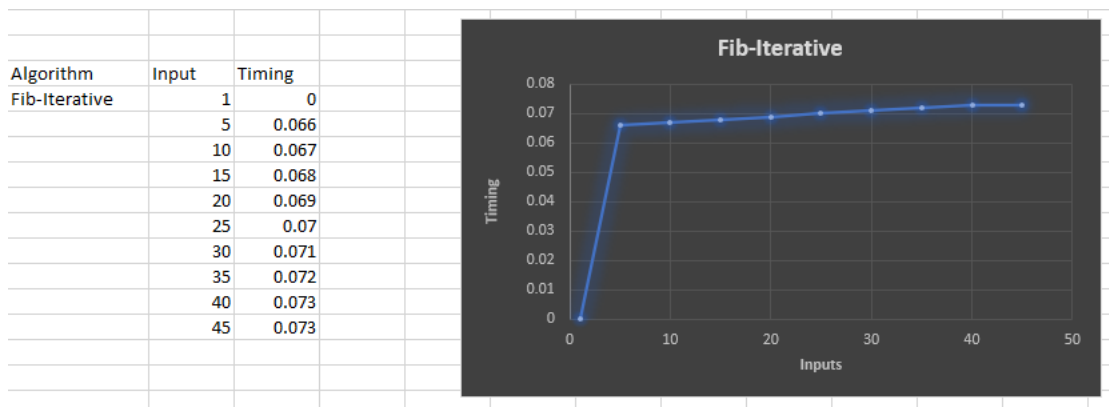
10. What method can be used to address recursive algorithms that excessively recompute?

Graphing - Fibonacci

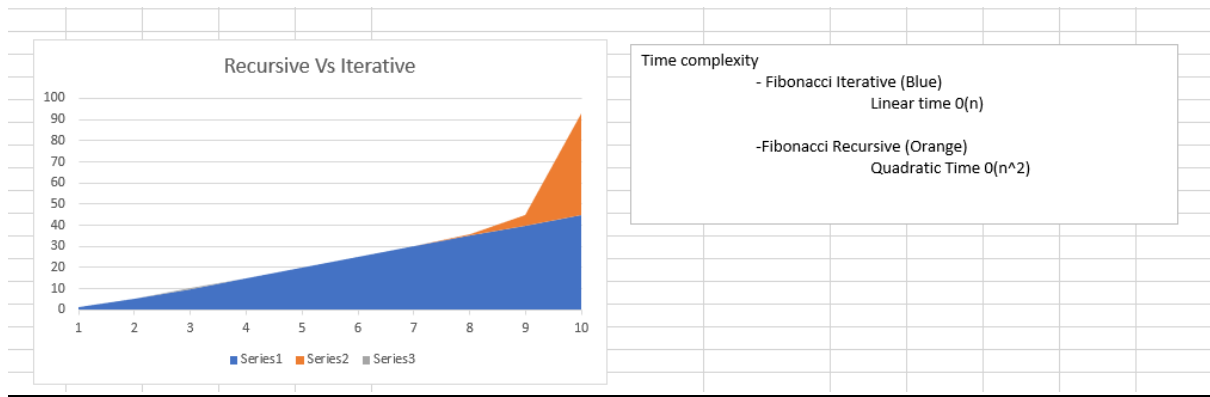


Time complexity for Fibonnaci Recursive solution – $O(2^n)$

Space Complexity – $O(n)$

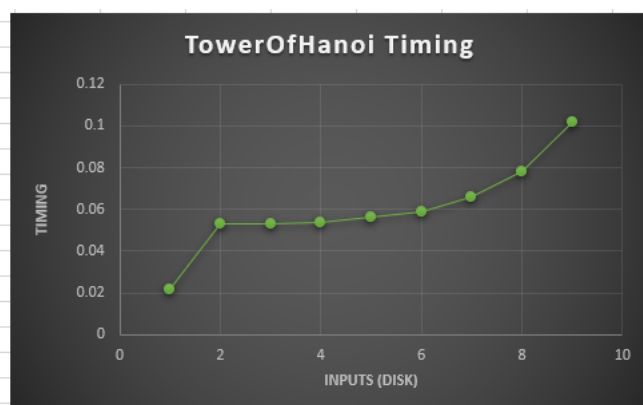


Time complexity for Fibonacci Iterative solution – $O(n)$



Graphing - Tower of Hanoi

Algorithm Inputs	timing	NumOfMoves
TowerOfH	1	0.022
	2	0.053
	3	0.053
	4	0.054
	5	0.056
	6	0.059
	7	0.066
	8	0.078
	9	0.102



Time complexity for Tower of Hanoi Recursive solution – $O(2^n)$

Space complexity for Tower of Hanoi Recursive solution