WHY IT'S A BAD IDEA TO USE RECURSION IN FINDING FIBONACCI OF A NUMBER

The recursive (a function that calls itself) solution to finding Fibonacci of a number has an exponential time complexity O(2^n) as opposed to the iterative solution which has a linear time complexity O(n). Algorithms with exponential time complexity are bad solutions to problems. This is because, for an increase by one in the number we want to find Fibonacci for, there is an exponential increase in the time the function takes to execute, as a result of multiple functions which are still on the stack, making it a bad solution that may end up freezing your machine for relatively larger numbers. However, this can be greatly improved using **memoization**(optimization technique used primarily to speed up computer programs by storing the results of expensive function calls and returning the cached result when the same inputs occur again).