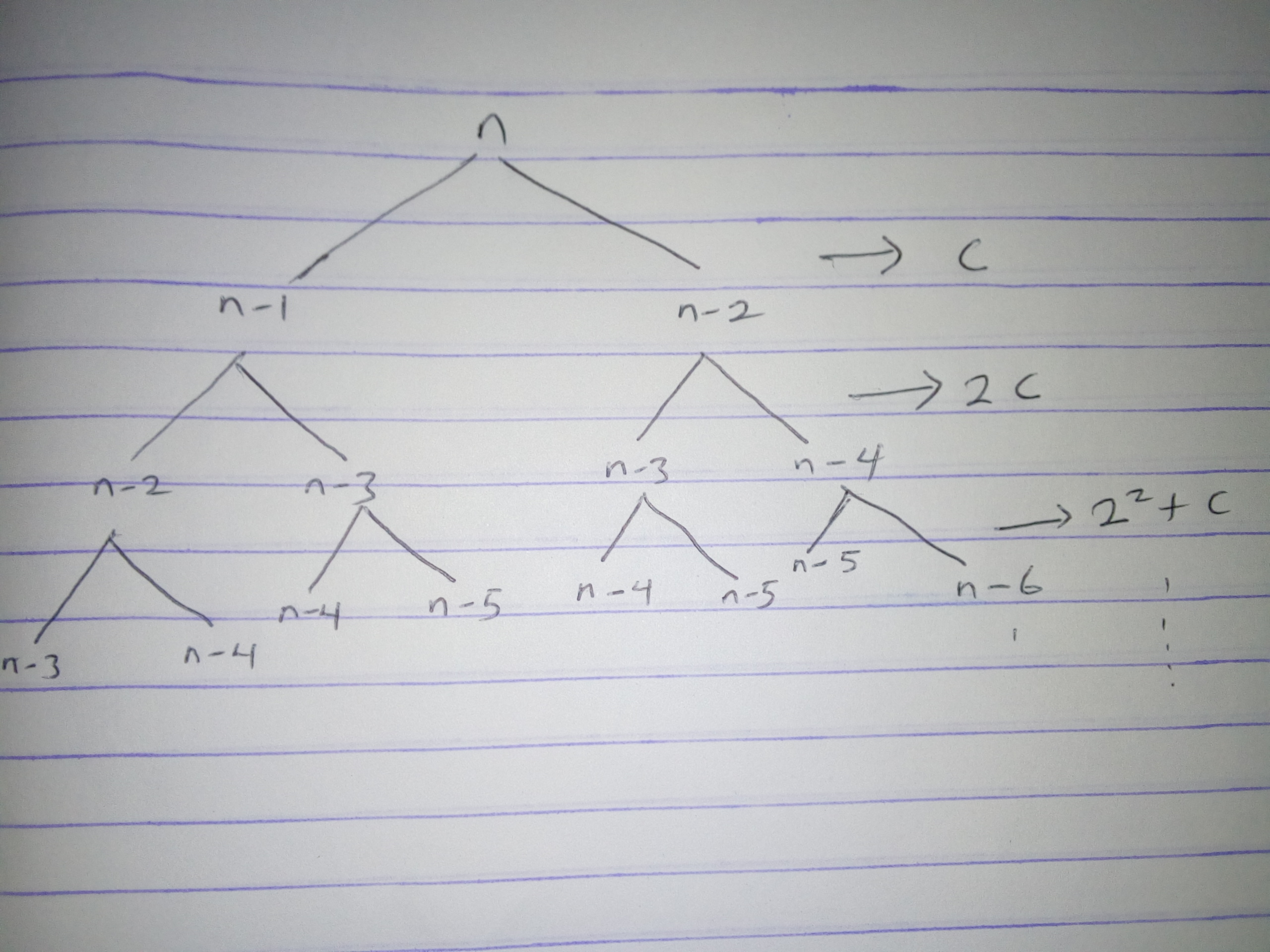
Using recursion is a bad idea to find the Fibonacci of a number based on time complexity. Recursion might be good when dealing with finding smaller terms in the sequence like the 15th term, 20th term or even 25th. But as the number increases the time to compute increases, making it extremely slow; take for instance a recursion to find the 1000th term, this will take a bit of time to compute compared to using iteration which is a lot faster. This can be illustrated by calculating the big O of a recursion:

T(n) = T(n-1) + T(n+2) + c



T(n) <= C + 2C + 22C + 23c….

T(n) <= C{ 1 + 22+23….+2n}

Taking the upper bond:

T(n) <= 2n

T(n) = O (2n)

That is, time complexity of a recursive Fibonacci algorithm is exponential while that of an iteration is linear {O (n)}. This implies that Fibonacci algorithm written to run iteratively will be faster than that which is written recursively.