Runtime Analysis

1) Time Complexity:

The fibonacci function makes two recursive calls for each input n:

fibonacci(n - 1) and fibonacci(n - 2).

This leads to a binary tree of recursive calls, where the number of nodes grows exponentially with n.

Therefore, the time complexity is O (2^n)

2) Space Complexity:

The space complexity is determined by the call stack.

The maximum depth of the recursion tree is n, so the call stack can grow up to n frames.

Therefore, the space complexity is O (n).