Kamil Gebski

1512307

Lab repository:

https://github.com/kamScripts/CT/tree/main/lab1

Number of function calls for n in range 20-35 with step of 5.

A black background with white text

AI-generated content may be incorrect.

Test results

## Reflection Questions – Answers

1. The main bottleneck of the naïve recursive approach are two recursive calls to calculate the nth number of the Fibonacci sequence. The Two recursive calls raise number of function calls exponentially.
2. The memoised version reduces the number of function calls significantly, as function at first checks if elements are in the cache. If elements are not in the cache, they are added to the memory, reducing the time of finding next numbers of the sequence.
3. The trade-off is a raising space complexity. The memorized version is faster but requires more memory for the interemediate values.
4. Bi-search Alghorithm
5. When running tests in the reverse order, naïve version gives the same results. Whereas the first calculation of memorized version requires significantly more calls, finding smaller elements of the sequence requires only one call as all higher values are already in the memory.

A black background with white text

AI-generated content may be incorrect.

Reverse tests