Expt 6 Div A Batch C

David Daniels 2023300038

DATE

In this experiment, I have implemented matrix chain multiplication algorithm using Dynamic programming. It also implements matrix multiplications using regular and strassens algorithm. It is done with C++

Time complexity

1) The time complexity of Strassens and Regular matrix multiplication is same i.e $O(n^3)$. But due to the divide and conquer operation in strassens, instead of 8 multiplications for the trivial state, we get 7. This will result in large amounts of time saved for ~~a~~ ~~which~~ ~~can~~ However, due to overhead in recursion, it is slower for small matrices.

2) The Matrix chain multiplication, has a time complexity of $O(n^3)$, It also has Space complexity of $O(n^2)$ but in the 2D array generated, we are only using half of it. Here, we have used bottom up dp, as we first check the matrices with distance 1, then dist 2 .. etc from each other. This will be faster than top down approach due to overhead costs even though the logic is the same. ~~Dyna~~ DP helps in reducing the number of unnecessary calls to find cost function