169. Given two 2×2 Matrices A and B

A=(1 7 B=(1 3

3 5) 7 5)

Use Strassen's matrix multiplication algorithm to compute the product matrix C such that C=A×B.

Test Cases:

Consider the following matrices for testing your implementation:

Test Case 1:

A=(1 7 B=(6 8
3 5), 4 2)

Expected Output:
C=(18 14
62 66)

Code:

```
def strassen_matrix_multiply(A, B):
    a11, a12, a21, a22 = A[0][0], A[0][1], A[1][0], A[1][1]
    b11, b12, b21, b22 = B[0][0], B[0][1], B[1][0], B[1][1]
    m1 = a11 * (b12 - b22)
    m2 = (a11 + a12) * b22
    m3 = (a21 + a22) * b11
    m4 = a22 * (b21 - b11)
    m5 = (a11 + a22) * (b11 + b22)
    m6 = (a12 - a22) * (b21 + b22)
    m7 = (a11 - a21) * (b11 + b12)
    c11 = m5 + m4 - m2 + m6
    c12 = m1 + m2
    c21 = m3 + m4
    c22 = m5 + m1 - m3 - m7
    C = [
        [c11, c12],
        [c21, c22]
    ]
    return C
A = [[1, 7], [3, 5]]
B = [[6, 8], [4, 2]]
C = strassen_matrix_multiply(A, B)
print("Result Matrix C:")
for row in C:
    print(row)
```

output:

```
PS C:\Users\karth>
PS C:\Users\karth> & C:/Users/karth/AppData/Local/Programs/Python/Python312/python.exe c:/Users/karth/OneDrive/Desktop/csa0863_karthik/PROBLEM.py
Result Matrix C:
[34, 22]
[38, 34]
PS C:\Users\karth>
```

Time complexity:$f(n)=o(2^n)$