

170. Given two 2×2 Matrices A and B

$A = \begin{pmatrix} 1 & 7 \\ 3 & 5 \end{pmatrix}$ $B = \begin{pmatrix} 1 & 3 \\ 7 & 5 \end{pmatrix}$

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

Test Cases:

Consider the following matrices for testing your implementation:

Test Case 1:

$A = \begin{pmatrix} 1 & 7 \\ 3 & 5 \end{pmatrix}$, $B = \begin{pmatrix} 6 & 8 \\ 4 & 2 \end{pmatrix}$

Expected Output:

$C = \begin{pmatrix} 18 & 14 \\ 62 & 66 \end{pmatrix}$

PROGRAM :-

```
def strassen_matrix_multiply(A, B):
```

```
    if len(A) == 2:
```

```
        a, b, c, d = A[0][0], A[0][1], A[1][0], A[1][1]
```

```
        e, f, g, h = B[0][0], B[0][1], B[1][0], B[1][1]
```

```
        p1 = a * (f - h)
```

```
        p2 = (a + b) * h
```

```
        p3 = (c + d) * e
```

```
        p4 = d * (g - e)
```

```
        p5 = (a + d) * (e + h)
```

```
        p6 = (b - d) * (g + h)
```

```
        p7 = (a - c) * (e + f)
```

```
        C = [[p5 + p4 - p2 + p6, p1 + p2], [p3 + p4, p1 + p5 - p3 - p7]]
```

```
        return C
```

```
    else:
```

```
        return "Input matrices are not 2x2."
```

Test Case

```
A = [[1, 7], [3, 5]]
```

```
B = [[6, 8], [4, 2]]
```

```
C = strassen_matrix_multiply(A, B)
```

```
print(C)
```

OUTPUT:-

```
[[34, 22], [38, 34]]
```

```
=== Code Execution Successful ===
```

TIME COMPLEXITY:- $7T(n/2) + O(n^2)$.