

83. Strassen's matrix multiplication

PROGRAM:-

```
import numpy as np
import time
```

```
def strassen_matrix_multiplication(A, B):
    n = len(A)
```

```
    if n == 1:
        return A * B
```

```
    # Splitting matrices into quadrants
```

```
    mid = n // 2
    A11 = A[:mid, :mid]
    A12 = A[:mid, mid:]
    A21 = A[mid:, :mid]
    A22 = A[mid:, mid:]
    B11 = B[:mid, :mid]
    B12 = B[:mid, mid:]
    B21 = B[mid:, :mid]
    B22 = B[mid:, mid:]
```

```
    # Computing the 7 products using Strassen's formulas
```

```
    M1 = strassen_matrix_multiplication(A11 + A22, B11 + B22)
    M2 = strassen_matrix_multiplication(A21 + A22, B11)
    M3 = strassen_matrix_multiplication(A11, B12 - B22)
    M4 = strassen_matrix_multiplication(A22, B21 - B11)
    M5 = strassen_matrix_multiplication(A11 + A12, B22)
    M6 = strassen_matrix_multiplication(A21 - A11, B11 + B12)
    M7 = strassen_matrix_multiplication(A12 - A22, B21 + B22)
```

```
    # Combining the results to get the final quadrants
```

```
    C11 = M1 + M4 - M5 + M7
    C12 = M3 + M5
    C21 = M2 + M4
    C22 = M1 - M2 + M3 + M6
```

```
    # Combining the quadrants into a single matrix
```

```
    C = np.vstack((np.hstack((C11, C12)), np.hstack((C21, C22))))
```

```
    return C
```

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

```
    start_time = time.time() # Start time measurement
```

```
    C = strassen_matrix_multiplication(A, B) # Perform Strassen's matrix multiplication
```

```
    end_time = time.time() # End time measurement
```

```
    elapsed_time = end_time - start_time
```

```

    return C, elapsed_time

# Example usage
n = 4 # Size of the matrix (must be a power of 2)
A = np.random.randint(0, 10, (n, n))
B = np.random.randint(0, 10, (n, n))

result, execution_time = find_strassen_time(A, B)

print("Matrix A:")
print(A)
print("Matrix B:")
print(B)
print("Result of Strassen's Matrix Multiplication:")
print(result)
print(f"Execution time: {execution_time:.10f} seconds")

```

OUTPUT:-

```

Matrix A:
[[3 6 6 9]
 [1 2 3 6]
 [8 7 0 1]
 [9 3 7 9]]
Matrix B:
[[0 4 8 0]
 [3 9 4 7]
 [8 6 9 8]
 [8 6 1 8]]
Result of Strassen's Matrix Multiplication:
[[138 156 111 162]
 [ 78  76  49  86]
 [ 29 101  93  57]
 [137 159 156 149]]
Execution time: 0.0002596378 seconds

=== Code Execution Successful ===

```

TIME COMPLEXITY:- $O(n^3)$