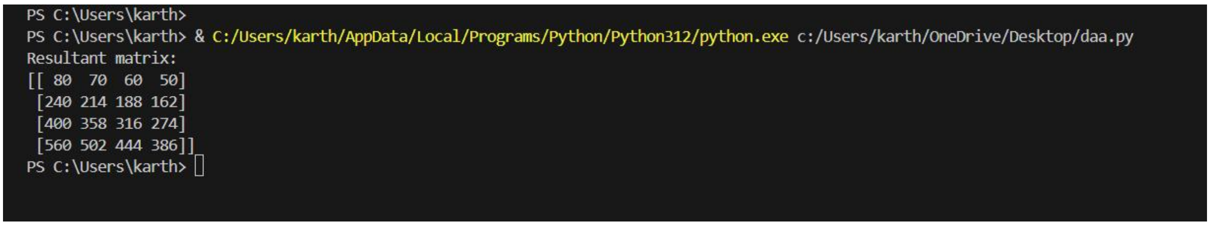
5. Strassens matrix multiplication.

Code:

```
import numpy as np
def strassen(A, B):
    n = A.shape[0]
    if n == 1:
        return A * B
    else:
        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:]
        M1 = strassen(A11 + A22, B11 + B22)
        M2 = strassen(A21 + A22, B11)
        M3 = strassen(A11, B12 - B22)
        M4 = strassen(A22, B21 - B11)
        M5 = strassen(A11 + A12, B22)
        M6 = strassen(A21 - A11, B11 + B12)
        M7 = strassen(A12 - A22, B21 + B22)
        C11 = M1 + M4 - M5 + M7
        C12 = M3 + M5
        C21 = M2 + M4
        C22 = M1 - M2 + M3 + M6
        C = np.vstack((np.hstack((C11, C12)), np.hstack((C21, C22))))
        return C
A = np.array([[1, 2, 3, 4],
          [5, 6, 7, 8],
          [9, 10, 11, 12],
          [13, 14, 15, 16]])
B = np.array([[16, 15, 14, 13],
          [12, 11, 10, 9],
          [8, 7, 6, 5],
          [4, 3, 2, 1]])
C = strassen(A, B)
print("Resultant matrix:")
print(C)
```

output:



```
PS C:\Users\karth>
PS C:\Users\karth> & C:/Users/karth/AppData/Local/Programs/Python/Python312/python.exe c:/Users/karth/OneDrive/Desktop/daa.py
Resultant matrix:
[[ 80  70  60  50]
 [240 214 188 162]
 [400 358 316 274]
 [560 502 444 386]]
PS C:\Users\karth>
```

Time complexity:

F(n)=o(n^2)