Strassen’s Matrix Multiplication

```
import numpy as np

def split(matrix):

   row, col = matrix.shape
   row2, col2 = row//2, col//2
   return matrix[:row2, :col2], matrix[:row2, col2:], matrix[row2:, :col2], matrix[row2:,
col2:]

def strassen(x, y):
   if len(x) == 1:
       return x * y

   a, b, c, d = split(x)
   e, f, g, h = split(y)

   p1 = strassen(a, f - h)
   p2 = strassen(a + b, h)
   p3 = strassen(c + d, e)
   p4 = strassen(d, g - e)
   p5 = strassen(a + d, e + h)
   p6 = strassen(b - d, g + h)
   p7 = strassen(a - c, e + f)

   c11 = p5 + p4 - p2 + p6
   c12 = p1 + p2
   c21 = p3 + p4
   c22 = p1 + p5 - p3 - p7

   c = np.vstack((np.hstack((c11, c12)), np.hstack((c21, c22))))

   return c
```