Divide and Conquer example: Matrix Multiplication

The normal procedure to multiply two $n \times n$ matrices requires n^3 time. We could improve the required running time by the following Strassen's matrix multiplication algorithm.
Given two $n \times n$ matrices A and B. Their product C is also an $n \times n$ matrix. Assuming that n is power of 2. We can divide each of A, B, and C into four $n/2 \times n/2$ matrices, rewriting the equation $A \times B = C$ as follows.

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$$

where let

$m_1 = (A_{12} - A_{22})(B_{21} + B_{22})$
$m_2 = (A_{11} + A_{22})(B_{11} + B_{22})$
$m_3 = (A_{11} - A_{21})(B_{11} + B_{12})$
$m_4 = (A_{11} + A_{12})B_{22}$
$m_5 = A_{11}(B_{12} - B_{22})$
$m_6 = A_{22}(B_{21} - B_{11})$
$m_7 = (A_{21} + A_{22})B_{11}$

Then compute the C_{ij} by the formulas

$C_{11} = m_1 + m_2 - m_4 + m_6$
$C_{12} = m_4 + m_5$
$C_{21} = m_6 + m_7$
$C_{22} = m_2 - m_3 + m_5 - m_7$

Assume that each scalar arithmetic operation takes constant time. Please write down the running time recurrence for the algorithm and derive its running time.