计算机本硕博2301 U202315763 王嘉乐

1. 不失一般性，设 $n=2^k$

$\therefore T(n) = 2T(\frac{n}{2}) + n\cdot\log_2 n$

$= 4T(\frac{n}{4}) + n\cdot\log_2 n + n\cdot\log_2\frac{n}{2}$

$= 8T(\frac{n}{8}) + n(\log_2 n + \log_2\frac{n}{2} + \log_2\frac{n}{4})$

$= \dots$

$= 2^k T(1) + n(\log_2 n + \log_2\frac{n}{2} + \log_2\frac{n}{2^2} + \dots + \log_2\frac{n}{2^{k-1}})$

$= 2^k T(1) + n\cdot\log_2\frac{n^k}{2^{k(k-1)/2}}$

$= 2^k T(1) + n(k\log_2 n - \log_2\frac{k(k-1)}{2})$

$= \Theta(n\log n)$

2. (1) $T(n) = 9T(n/3) + n$

$a=9 \quad b=3$

$\therefore \exists\varepsilon=1, \quad f(n) = n = O(n^{\log_b a-\varepsilon})$

$\therefore T(n) = \Theta(n^{\log_b a}) = \Theta(n^2)$

(2) $T(n) = T(2n/3) + 1$

$a=1 \quad b=3/2$

$\therefore f(n) = 1 = \Theta(n^{\log_{3/2} 1})$

$\therefore T(n) = \Theta(n^{\log_{3/2} 1}\log n) = \Theta(\log n)$

(3) $T(n) = 3T(n/4) + n\log n$

$a=3 \quad b=4$

$\therefore \exists\varepsilon, \quad f(n) = \Omega(n^{\log_4 3+\varepsilon})$

$\therefore T(n) = \Theta(f(n)) = \Theta(n\log n)$

3. $A=\begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix}$ $B=\begin{bmatrix}B_{11} & B_{12}\\ B_{21} & B_{22}\end{bmatrix}$ $C=\begin{bmatrix}C_{11} & C_{12}\\ C_{21} & C_{22}\end{bmatrix}$

$$P_1 = A_{11}(B_{12}-B_{22})$$
$$P_2 = (A_{11}+A_{12})B_{22}$$
$$P_3 = (A_{21}+A_{22})B_{11}$$
$$P_4 = A_{22}(B_{21}-B_{11})$$
$$P_5 = (A_{11}+A_{22})(B_{11}+B_{22})$$
$$P_6 = (A_{12}-A_{22})(B_{21}+B_{22})$$
$$P_7 = (A_{11}-A_{21})(B_{11}+B_{12})$$

验证：

$$C_{11} = P_5 + P_4 - P_2 + P_6$$
$$= (A_{11}B_{11} + \cancel{A_{11}B_{22}} + \cancel{A_{22}B_{11}} + \cancel{A_{22}B_{22}}) + (\cancel{A_{22}B_{21}} - \cancel{A_{22}B_{11}}) - (\cancel{A_{11}B_{22}} + \cancel{A_{12}B_{22}})$$
$$+ (A_{12}B_{21} + \cancel{A_{12}B_{22}} - \cancel{A_{22}B_{21}} - \cancel{A_{22}B_{22}})$$
$$= A_{11}B_{11} + A_{12}B_{21}$$

$$C_{12} = P_1 + P_2$$
$$= (A_{11}B_{12} - \cancel{A_{11}B_{22}}) + (\cancel{A_{11}B_{22}} + A_{12}B_{22})$$
$$= A_{11}B_{12} + A_{12}B_{22}$$

$$C_{21} = P_3 + P_4$$
$$= (A_{21}B_{11} + \cancel{A_{22}B_{11}}) + (A_{22}B_{21} - \cancel{A_{22}B_{11}})$$
$$= A_{21}B_{11} + A_{22}B_{21}$$

$$C_{22} = P_5 + P_1 - P_3 - P_7$$
$$= (\cancel{A_{11}B_{11}} + \cancel{A_{11}B_{22}} + \cancel{A_{22}B_{11}} + A_{22}B_{22}) + (\cancel{A_{11}B_{12}} - \cancel{A_{11}B_{22}}) - (\cancel{A_{21}B_{11}} + \cancel{A_{22}B_{11}})$$
$$- (\cancel{A_{11}B_{11}} + \cancel{A_{11}B_{12}} - \cancel{A_{21}B_{11}} - A_{21}B_{12})$$
$$= A_{21}B_{12} + A_{22}B_{22}$$

4. 算法~~曾~~思想：①划分：将数组划分为左右两部分
②递归求解左、右两部分中的最大子数组的和
③合并：求跨越中间的最大子数组和
④比较三个值，取最大的值.

伪代码：

```
MaxSubArray (A, low, high):
    if low == high
        return A[low]
    mid = (low + high)/2
    leftMax  = MaxSubArray (A, low, mid)
    rightMax = MaxSubArray (A, mid+1, high)
    crossMax = MaxCrossingSubArray (A, low, mid, high)
    return max (leftMax, rightMax, crossMax)

MaxCrossingSubArray (A, low, mid, high):
    leftSum = -∞
    sum = 0
    for i = mid downto low:
        sum += A[i]
        if sum > leftSum:
            leftSum = sum
    rightSum = -∞
    sum = 0
    for i = mid+1 to high:
        sum += A[i]
        if sum > rightSum:
            rightSum = sum
    return leftSum + rightSum
```

时间复杂度：

$T(n) = 2T(n/2) + n$

$a=2 \quad b=2 \quad d=1$

$\because a = b^d$

$\therefore T(n) = O(n^d \log n) = O(n\log n)$