Time Complexity of Merge Sort

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

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

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

$$T(n) = 2T(n/2) + n \longrightarrow (i)$$

$$T(n/2) = 2T(n/4) + \frac{n}{2} \longrightarrow (ii)$$

$$T(n/4) = 2T\left(\frac{n}{8}\right) + \frac{n}{4} \longrightarrow (iii)$$

Putting equ (ii) in equ (i)

$$T(n) = 2\left[2T\left(\frac{n}{4}\right) + \frac{n}{2}\right] + n$$

$$= 4T\left(\frac{n}{4}\right) + 2n$$

Putting equ (iii) in equ (ii)

$$T(n) = 8T\left(\frac{n}{8}\right) + 3n$$

$$= 2^k T\left(\frac{n}{2^k}\right) + kn$$

$$\frac{n}{2^k} = 1$$

$$n = 2^k$$

$$k = \log_2 n$$

$$T(n) = 2^k T\left(\frac{n}{2^k}\right) + kn$$

$$= 2^k T(1) + kn$$

$$= 2^k + kn$$

$$= n + n\log_2 n$$

$$T(n) = O(n + n\log_2 n)$$

$$\boxed{T(n) = O(n \log_2 n)}$$