$$T(n) = 2^1 T(n/2^1) + 1 \cdot n \quad \text{—— (1)}$$

(Backward Substitution method) $T(1) = 1$ ✓

put $n = n/2$ in (1) $T(n/2) = 2T(n/4) + n/2$

$$= 2[2T(n/4) + n/2] + n$$

put $n = n/4$ in (1) $T(n/4) = 2T(n/8) + n/4$

$$= 2^2 T(n/2^2) + 2n \quad \text{—— (2)}$$

$$= 2^2 [2T(n/8) + n/4] + 2n$$

$$= 2^3 T(n/2^3) + 3n \quad \text{—— (3)}$$

$$= 2^4 T(n/2^4) + 4n \quad \text{—— (4)}$$

$$\vdots$$

$$T(n) = 2^k T(n/2^k) + kn \quad \text{—— (5)}$$

$$\frac{n}{2^k} = 1 \Rightarrow 2^k = n \Rightarrow k = \log_2 n$$

$$= n\, T(1) + n \cdot \log n$$

$$T(n) = n + n\log n = O(n\log n) \checkmark$$

merge sort (All cases)

(Quick sort best)

Rec-Binary search

$$T(n) = T(n/2^1) + 1 \quad \text{—— (1)}$$

$$T(n) = T(n/4) + 1 + 1$$

$$= T(n/2^2) + 2 \quad \text{—— (2)}$$

$$= T(n/8) + 3$$

$$= T(n/2^3) + 3 \quad \text{—— (3)}$$

$$\vdots$$

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

$$= T(1) + \log n = \log n + 1 = O(\log n)$$

$$T(1) = 1$$

$$T(n/2) = T(n/4) + 1$$

$$T(n/4) = T(n/8) + 1$$

$$n/2^k = 1 \Rightarrow 2^k = n$$

$$\Rightarrow k = \log n$$