

Binary Search

$$T(n) = \begin{cases} 1 & n = 1 \\ T(n/2) + 1 & n > 1 \end{cases}$$

Substitution method:

$$T(n) = T(n/2) + 1 \rightarrow \textcircled{i}$$

$$T(n) = [T(n/2) + 1] + 1$$

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

$$T(n) = T(n/2^3) + 3 \rightarrow \textcircled{ii}$$

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

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

$$\therefore n = 2^k \Rightarrow k = \log n$$

$$\text{from (iii)} \Rightarrow T(n) = T\left(\frac{n}{2^k}\right) + k$$

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

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