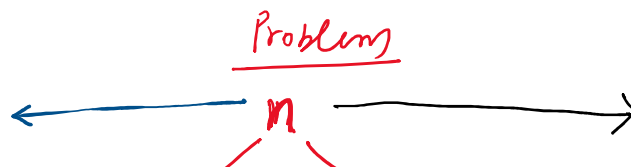


Q2

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

Input:

$$n = n/2^0$$



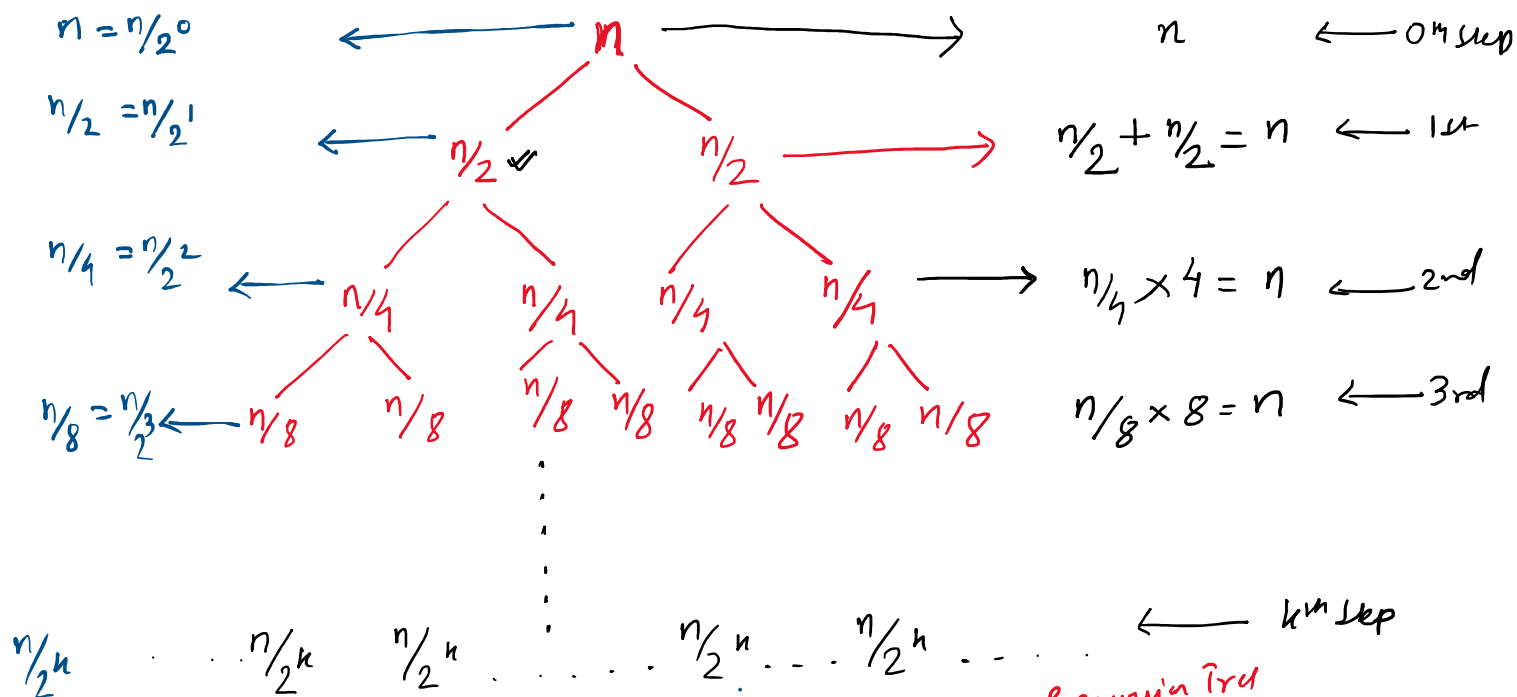
Cost

n

← 0th step

Total cost

$$= \underbrace{n + n + n + \dots + n}$$



$$= \underbrace{n + n + n + \dots + n}_{k \text{ no of } n}$$

$$= n \times k$$

$$= n \log_2 n$$

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

Recursion stops when input becomes $n/2^k = 1$

$$\Rightarrow n = 2^k$$

$$\Rightarrow \log_2 n = \log_2 2^k = k \times \log_2 2$$

$$\Rightarrow \log_2 n = k$$

$$\log_a a = 1$$