ANALISI di QUICKSORT

$$T(n) = \begin{cases} O(1) & n \leq 1 \\ T(q) + T(n-1-q) + \underline{\Theta(n)} & n > 1 \end{cases}$$

↳ costo di Partition

→ se ad ogni livello di ricorsione si ha una ripartizione completamente sbilanciata:

$$T(n) = T(0) + T(n-1) + \Theta(n)$$

$$\rightsquigarrow T(n) = T(n-1) + \Theta(n)$$

$$T(n) = \Theta(n^2)$$

→ se ad ogni livello di ricorsione si ha una ripartizione completamente bilanciata

$$T(n) = T\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + T\left(\left\lceil \frac{n}{2} \right\rceil - 1\right) + \Theta(n)$$

$$\rightsquigarrow T(n) = 2T(n/2) + \Theta(n)$$

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

Si può dimostrare che

1) Il tempo di esecuzione di QuickSort è $O(n^2)$

→ si studia la ricorrenza

$$T(n) = \max_{0 \leq q \leq n-1} \{T(q) + T(n-q-1)\} + \Theta(n)$$

e si dimostra che $T(n) \leq c \cdot n^2$

2) Il tempo di esecuzione di QuickSort è $\Omega(n \log n)$

→ si studia la ricorrenza

$$T(n) = \min_{0 \leq q \leq n-1} \{T(q) + T(n-q-1)\} + \Theta(n)$$

e si dimostra che $T(n) \geq c \cdot n \cdot \log n$