Hands on 6

Quick Sort time complexity

Avg case =

Array gets divided into 2 parts of size k & $(N-K)$

$$T(N) = T(N-K) + T(K) = \frac{1}{N}\left[\sum_{i=1}^{N-1} T(i) + \sum_{i=1}^{N-1} T(N-i)\right]$$

$$T(N) = \frac{2}{N}\sum_{i=1}^{N-1} T(i)$$

$$N(T(N)) = 2\sum_{i=1}^{N-1} T(i) \quad \text{— (1)}$$

$$(N-1)(T(N-1)) = 2\sum_{i=1}^{N-2} T(i) \quad \text{— (2)}$$

Subtracting (2) from (1)

$$N(T(N)) - (N-1)(T(N-1)) = 2T(N-1) + N^2 C - (N-1)^2 C^2$$

$$NT(N) = T(N-1)(2+N-1) + \cancel{C} + 2NC - \cancel{C}$$

$$= (N+1)\,T(N-1) + 2NC$$

$$\frac{\cancel{N}\,T(N)}{\cancel{N}(N+1)} = \frac{\cancel{(N+1)}\,T(N-1)}{N\cancel{(N+1)}} + \frac{2\cancel{N}C}{\cancel{N}(N+1)} \quad \text{— (3)}$$

if $N = N-1$

$$\frac{T(N-1)}{N} = \frac{T(N-2)}{(N-1)} + \frac{2C}{N}$$

$$= \frac{T(N)}{N+1} = \frac{T(N-2)}{(N-1)} + \frac{2C}{(N+1)} + \frac{2C}{N}$$

We can get the value of $T(N-2)$ by replacing N by $(N-2)$ in ③

$$T(N) = 2C \log_2 N (N+1)$$

$$T(N) = \log_2 N * (N+1)$$

$$T(N) = N \log_2 N + \log_2 N$$

$$\therefore T(N) = O(N \log_2 N)$$