

بررسی مرتب سازی سریع (sort Quick)

$$W(n) = n(n-1)/2 \rightarrow w(n) \leq n(n-1)/2 = \theta(n^2)$$

برای پارتیشن $T(n) = n-1$

$$A(n) = 1/n \sum A(p-1) + A(n-p) + n-1$$

$$= 1/n \sum A[(p-1) + A(n-p)] + n-1$$

$$A(0) + A(n-1) + A(1) + A(n-2) + \dots + A(n-2) + A(n-1) + A(1) + A(0)$$

$$= 2/n \sum A(p-1) + n-1$$

$$nA(n) = 2 \sum A(p-1) + n(n-1)$$

$$(n-1)A = 2 \sum A(p-1) + (n-1)(n-2)$$

$$nA(n) - (n-1)A(n-1) = 2A(n-1) + 2(n-1)$$

$$nA(n) - (n+1)A(n-1) = 2(n-1)$$

$$A(n)/n+1 = a(n-1)/n + 2(n-1)/n(n+1)$$

$$N = A(n) / n+1 \rightarrow An = an-1 + 2(n-1) / n(n+1)$$

$$n-1 \sim n+1$$

$$\sim an = an-1 + 2/n$$

$$= an-2 + 2/n-1 + 2/n$$

$$= an-3 + 2/n-2 + 2/n-1 + 2/n$$

$$\rightarrow an \sim 2 \lg n$$

$$A(n) / n+1 = 2 \ln n \rightarrow A(n) = 2(n+1) \lg n$$

$$= 2(n+2) \lg n * \ln 2$$

$$= 1.38(n+1) \lg n$$

$$= \theta(n \lg n)$$

در روش مرتب سازی Quick sort تعداد متوسط مقایسه ها برابر با $n \log n$ و تعداد تعویضها برابر با $(n/6) \log$ است.

این مقدار در مقایسه با سایر الگوریتمهایی که تاکنون بررسی شده اند بسیار کمتر است . لذا الگوریتم **sort Quick** از کلیه

الگوریتمهای مرتب سازی که تاکنون بررسی شده اند ، بهتر است.

استاد سهمی که این الگوریتم **Quick Sort** به روشهایی که کد نویسی شده است تک تک آنها به صورت تریس انجام شده است.

با تشکر فراوان

حمید امید