

بررسی best case برای Quick Sort به ازای $n = ۱۶$

کتابخانه صبر

$$T(n) = T(i) + T(n-i-1) + \Theta(n) \quad \text{بعد می دانیم } T(n) \text{ از رابطه}$$

بهترین می دانیم best case آن $\Theta(n \lg n)$ است.

لح. به ازای $i = \frac{n}{2}$ و best case برای Quick Sort به بهترین می دانیم.

$$T(n) = \left[T\left(\frac{n}{2}\right) + T\left(\frac{n}{2} - 1\right) + \Theta(n) \right] \in \underbrace{\Theta(2T\left(\frac{n}{2}\right) + \Theta(n))}_{\text{این وقت برای Merge Sort این به ۳ که از } \Theta(n \lg n) \text{ است.}}$$

این وقت برای Merge Sort این به ۳ که از $\Theta(n \lg n)$ است.

اکنون مسئله را برای n های کوچک ببزرگ و پس بازگشتی حل می کنیم. استراتژی این است که

عدد میانه (یا یکی از دو عدد وسط، برای n های زوج) در آخرین خانه و باقی عناصر، تمام کمتر از درخت

چپ جایگاه اصلی میانه در سمت راست، خود جایگاه اصلی میانه قرار گیرند و در نتیجه آنها

جایگاه جایی، جایگاه جایی عنصر میانه است.

* $n = ۳$:

۱	۳	۲
---	---	---

با عنصر آخر swap می شود



* $n = ۴$:

۱	۴	۳	۲
---	---	---	---

* $n = 8$:

۱	۳	۲	۶	۴	۸	۷	۴
---	---	---	---	---	---	---	---

↓
جایگاه عنصر آخر

* توضیح : این آرایه‌ها را از فراخوان partition به صورت زیر در می‌آوریم :

۱	۳	۲	۴	۴	۸	۷	۶
---	---	---	---	---	---	---	---

بازگشتی برای $n=3$

بازگشتی برای $n=4$

↓

این قسمت با افزودن ۴ به هر کدام از عناصر در فرم $n=4$ به دست آمده است.

۱	۳	۲	۶	۴	۷	۴
---	---	---	---	---	---	---

بازگشتی برای $n=3$

بازگشتی برای $n=3$

* $n = 7$:

* $n = 14$:

۱	۳	۲	۶	۴	۷	۴	۱۲	۹	۱۱	۱۰	۱۴	۱۳	۱۶	۱۸	۸
---	---	---	---	---	---	---	----	---	----	----	----	----	----	----	---

که پس از فراخوان partition به شکل زیر در می‌آوریم :

۱	۳	۲	۶	۴	۷	۴	۸	۹	۱۱	۱۰	۱۴	۱۳	۱۶	۱۸	۱۲
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----

بازگشتی برای $n=7$

بازگشتی برای $n=8$

(با افزودن ۸ واحد به هر خانه)