جلسه چهارم

13:15 Tuesday, September 28, 2021

موضوع: تحلیل مجانبی و روابط بازگشتی

Θ و Ω و O

$f(n) = O(g(n))$

مثلاً $f(n) = 4n^2 + 5n + 7 = O(n^2)\ \ O(n^3)\ \ O(n^4)$

$= \Omega(n^2)\ \ \Omega(n)\ \ \Omega(n \log n)\ \ \Omega(\log n)$

$= \Theta(n^2)$

$\log_2^{n!} = \Theta(n \log n) \longrightarrow O(n \log n)$

$\Omega(n \log n)$

$\log_{\underline{x}}^{n} = \Theta(\log_{\underline{2}}^{n})$

یه مقدار ثابت

جهت یادآوری:

$f(n) = O(g(n)) : \exists\, n_0, c \ \forall n \geq n_0 \ \ f(n) \leq c\, g(n)$

$\underset{\mathbb{Z}}{n_0},\ \underset{>0}{c}$

$$\boxed{c \uparrow \text{ به طوری که } n_0 = 1}$$

1. $f(n) = O(g(n)) \iff g(n) = \Omega(f(n))$

2. $f_1(n) = O(g_1(n))$ $\quad f_1(n) \times f_2(n) = O(g_1(n) \times g_2(n))$

2. $f_1(n) = O(g_1(n))$ $f_1(n) \times f_2(n) = O(g_1(n) \times g_2(n))$

$f_2(n) = O(g_2(n))$ $f_1(n) + f_2(n) = O(\max[g_1(n), g_2(n)])$

$$x + y \leq 2 \times \max[x, y]$$

ثابت O بیشترین استفاده را دارد.

$O(1)$ $O(\log n)$ $O(n)$ $O(n \log n)$ $O(n^2)$ $O(n^3)$ $O(n^k)$ $O(2^n)$

زمان ثابت

* ساده سازی محاسبه زمان اجرای الگوریتم.

مثال:

$O(1)$ [═══ $\rightarrow O(1)$

for(i:1 → n)
    ═══] $O(1)$] $O(n) \times O(1) = O(n)$

for(i:1 → n)
    for(j:1 → n)
        ═══] $O(1)$
        bin_search(1, n)] $O(\log n)$] $O(\log n)$] $O(n \log n)$

$O(n^2 \log n)$

$O(n)$ [seq_search(1, n)

$$O(n^2 \log n)$$

کش ، ۲) تحلیل روابط بازگشتی : روش جایگذاری ، استقرا ، درخت بازگشت

بخش ۲) تحلیل روابط بازگشتی : روش جایگذاری ، استقرا ، درخت بازگشت

← قضیه اصلی

مثال ۱) زمان اجرای کد زیر چقدر است؟ (جستجوی دودویی)

```
bin_search(low, high){
    if(low > high)
        return
    mid = (low+high)/2
    if( A[mid] > x)
        bin....(low, mid-1)
    if( A[mid] < x )
        bin....(mid+1, high)
    return (mid)
}
```

$T(0)$ ، $O(1)$ ، $O(1)$ ، $T(\frac{n}{2})$ ، $O(1)$

$T(n)$: زمان اجرای الگوریتم در یک آرایه با اندازه n

$$T(n) = T(\frac{n}{2}) + O(1)$$

$$T(0) = O(1)$$

$$T(n) = T(\frac{n}{2}) + \boxed{O(1)}$$

$$= T(\frac{n}{4}) + O(1) + \boxed{O(1)}$$

$$= T(\frac{n}{8}) + O(1) + O(1) + O(1)$$

$$\vdots$$

$$= T(1) + O(1) + \cdots + O(1)$$

$$= T(0) + O(1) + \cdots\cdots + O(1)$$

$$= O(1) + O(1) + \cdots + O(1)$$

$$= O(\log n) \times O(1) = O(\log n)$$

مثال ۲. $T(n) = T(\frac{n}{2}) + O(n)$ $T(1) = O(1)$

$$T(n) = T(\tfrac{n}{۲}) + O(n)$$

$$= T(\tfrac{n}{۴}) + \underbrace{O(\tfrac{n}{\boxed{۲}})} + O(n)$$

$$= T(\tfrac{n}{۸}) + O(n) + O(n) + O(n)$$

$$\vdots$$

$$= \underbrace{T(1)}_{O(1)} + \underbrace{O(n) + O(n) + \dots + O(n)}_{\log n \times O(n)}$$

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

$$T(1) = O(1)$$

$$\exists c_1 : T(1) \leqslant c_1$$

$$T(n) = T(\tfrac{n}{۲}) + O(n)$$

$$\exists c_۲ \quad T(n) \leqslant T(\tfrac{n}{۲}) + c_۲ n$$

$$T(n) \leqslant \boxed{T(\tfrac{n}{۲})} + c_۲ n$$

$$\leqslant \boxed{T(\tfrac{n}{۴}) + c_۲ \tfrac{n}{۲}} + c_۲ n$$

$$\leqslant T(\tfrac{n}{۸}) + c_۲ \tfrac{n}{۴} + c_۲ \tfrac{n}{۲} + c_۲ n$$

$$\vdots$$

$$\leqslant T(1) + c_۲ \times ۲ + c_۲ \times ۴ + \dots + c_۲ \tfrac{n}{۲} + c_۲ n$$

$$\leqslant c_1 + c_۲ \left[۲ + ۴ + \dots + \tfrac{n}{۲} + n \right]$$

$\leq 2n$

$\leq \underbrace{c_1} + c_2 n$

$= O(n)$ ✓

مثال ۳) مرتب سازی ادغامی یا Mergesort .

هدف : مرتب کردن آرایه A با اندازه n

زیر مسأله (Merge) : دو آرایه A_1 و A_2 به صورت مرتب‌شده داده شده است . A_1 و A_2 را ادغام کنید .

۳	۶	۸	۱۰

۲	۴	۵	۱۱

۲	۳	۴	۵	۶	۸	۱۰	۱۱

۲	۳	۴	۵	۶	۸	۱۰	۱۱

$|A_1| + |A_2| = n$

زمان الگوریتم ادغام $O(n)$.

مرتب سازی ادغامی :

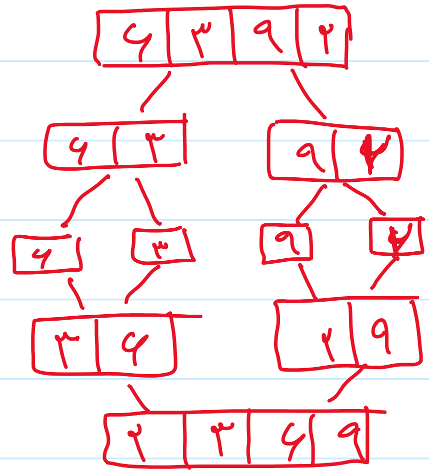
A_1	A_2

۱- آرایه A را به دو قسمت مساوی تقسیم کنید : A_1 و A_2

۲- A_1 و A_2 را به صورت بازگشتی مرتب کنید

۳- A_1 و A_2 را ادغام کنید .

```
merge_sort(low, high) {
O(1)      if (low >= high)
              return
O(1)      mid = (low + high)/2
T(n/2)    merge_sort(low, mid)
T(n/2)        "      (mid+1, high)
O(n)      merge([low, mid], [mid+1, high])
}
```



$$T(n) = 2T\left(\frac{n}{2}\right) + O(n) \qquad T(1) = O(1)$$

$$T(n) \leqslant 2T\left(\frac{n}{2}\right) + \underline{c_2\, n} \qquad T(1) \leqslant c_1$$

استقرا : نشان می‌دهیم $\boxed{T(n) \leqslant c_3\, n \log n}$ که در آن $c_3 = c_2 + c_1$

$\boxed{\text{n توانی از ۲}}$

پایه = $n = 2$

$$T(2) \leqslant 2T(1) + 2c_2$$

$$\leqslant 2(c_1 + c_2) = \underline{2(c_1 + c_2)} \times \underbrace{\log_2^2}_{1}$$

$$2 \times c_3 \times \log_2^2$$

فرض : گزاره به ازای همه اعداد کمتر از n صحیح است. برای n داریم :

$$T(n) \leqslant 2\,T\left(\frac{n}{2}\right) + c_2\, n$$

$$T(n) \leqslant ۲\,T\left(\frac{n}{۲}\right) + C_۲\, n$$

$$\leqslant \cancel{۲}\, C_۳\, \frac{n}{\cancel{۲}} \underbrace{\log \frac{n}{۲}}_{(\log n - 1)} + C_۲\, n$$

$$\leqslant C_۳\, n \left(\log n - 1\right) + C_۲\, n$$

$$\leqslant C_۳\, n \log n - C_۳\, n + C_۲\, n$$

$$\leqslant C_۳\, n \log n \qquad \square$$

$$T(n) \leqslant C_۳\, n \log n = O(n \log n) \qquad \square$$

$T(n) = \Theta(n \log n) \longleftarrow T(n) = \Omega(n \log n)$: Remark

---

کوچکترین ← _ _ _ _ _ _ _ _

_ _ _ .