



دانشگاه صنعتی اصفهان
دانشکده مهندسی برق و کامپیوتر

تکلیف دوم درس طراحی الگوریتم‌ها

نیم‌سال تحصیلی: بهار ۱۴۰۲

مدرس: دکتر محمدرضا حیدرپور

دستیاران آموزشی: مصطفی دریس‌پور - مجید فرهادی - محمدیاسین

کرباسیان - محمدرضا مزروعی - امیر منصوریان - امیرارسلان یآوری

۱ سکه قلبی

تعداد زیادی سکه که تعداد آن‌ها توانی از عدد 3 است و یک ترازوی دوکفه‌ای در اختیار داریم که می‌توان هر تعداد سکه را در هر کفه آن قرار داد. در میان این سکه‌ها دقیقاً یک سکه با وزنی متفاوت (کم‌تر یا بیش‌تر) وجود دارد. الگوریتمی با رویکرد تقسیم و غلبه برای یافتن این سکه قلبی با کم‌ترین تعداد مقایسه ارائه دهید. (۱۰ نمره)

۲ یگانه

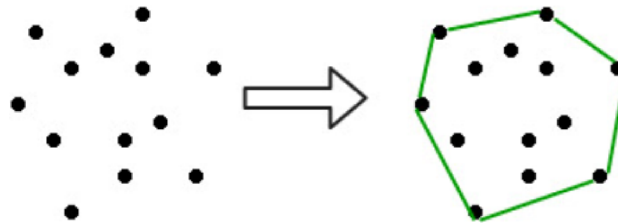
آرایه‌ای به طول فرد که همه عناصر آن به جز یکی، دو بار تکرار شده‌اند مفروض است. الگوریتمی با رویکرد تقسیم و غلبه برای یافتن این عنصر یگانه ارائه دهید. (۱۰ نمره)

$$A = [1, 5, 4, 8, 1, 8, 2, 2, 5]$$

$$Unique = 4$$

۳ تحدب

آرایه‌ای از مختصات تعدادی نقطه در صفحه مفروض است. الگوریتمی با رویکرد تقسیم و غلبه برای یافتن زیرمجموعه‌ای از این نقاط که یک چندضلعی محدب را تشکیل داده و شامل تمام نقاط دیگر است ارائه دهید. (۱۰ نمره)



۴ میانه

الگوریتمی با رویکرد تقسیم و غلبه برای یافتن میانه ادغام دو آرایه مرتب ارائه دهید. (۱۰ نمره)

$$A = [1, 12, 15, 26, 38]$$

$$B = [2, 13, 17, 30]$$

$$Merge(A, B) = [1, 2, 12, 13, 15, 17, 26, 30, 38]$$

$$Median = 15$$

۵ روابط بازگشتی

رفتار مجانبی توابع زیر را بیابید. (۲۰ نمره)

- $T(1) = 1, T(2) = 2, T(n) = 8T(n-1) - 15T(n-2)$
- $T(1) = 1, T(n) = T(\frac{n}{4}) + T(\frac{3n}{4}) + n^2$
- $T(2) = 1, T(n) = 4T(\lceil \sqrt{n} \rceil) + 1$
- $T(1) = 1, T(2^n) = 7T(2^{n-1}) + 2^n$

۶ کوئرا

به یکی از سوالات کوئرا پاسخ دهید. (۴۰ نمره)

۷ بهینه‌سازی

آرایه‌ای شامل n عدد صحیح و i و j به عنوان اندیس دو عنصر متفاوت از این آرایه مفروض است. تابع $f(i, j)$ به صورت زیر تعریف می‌شود.

$$f(i, j) = g(i, j)^2 + (i - j)^2$$

که در آن تابع $g(i, j)$ از برنامه زیر به دست می‌آید.

```
int g(int i, int j)
{
    int sum = 0;
    int min = (i < j) ? i : j;
    int max = (i > j) ? i : j;
    for (int k = min + 1; k <= max; k++)
    {
        sum += a[k];
    }
    return sum;
}
```

الگوریتمی از مرتبه زمانی $O(n \log n)$ برای محاسبه کمینه مقدار $f(i, j)$ ارائه دهید. (راهنمایی: این مسئله چه ارتباطی با مسئله Closest Pair دارد؟) (۲۰ نمره مازاد)

① اثر n سکه داشته باشیم برای پیدا کردن سکه با وزن متفاوت الگوریتم زیر را داریم:

هر بار سکه ها را به ۳ دسته تقسیم می کنیم و با مقایسه وزن این ۳ دسته، دسته با وزن کمتر را پیدا می کنیم (۲ عملیات مقایسه ای برای $n=3$ و $n=4$) سپس همین روند را برای دسته با وزن متفاوت اجرا می کنیم این کار را تا جایی ادامه می دهیم که به یک سکه متفاوت برسیم. با کمک توابع recursive داریم:

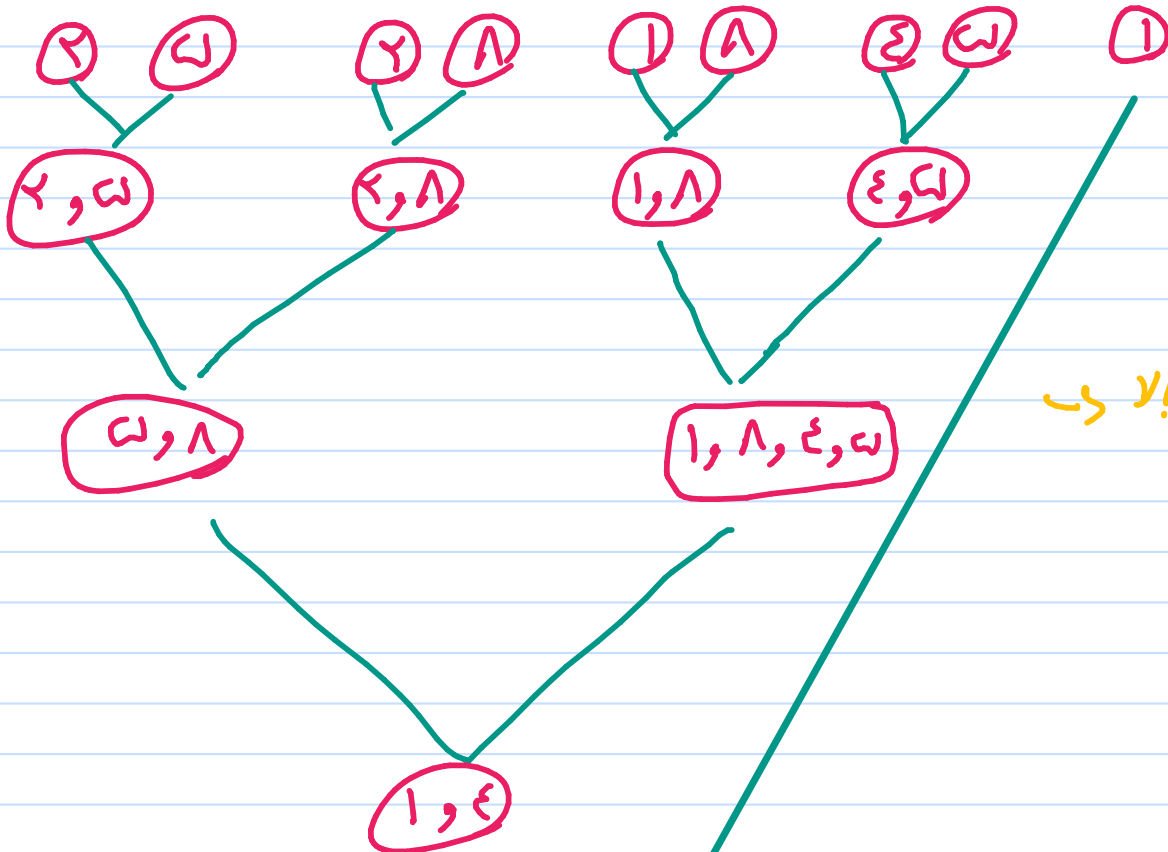
$$T(n) = T\left(\frac{n}{3}\right) + 2 \quad \text{تعداد مقایسه ها} \quad \text{باید: } T(1) = 1$$

$$\Rightarrow T(n) = T\left(\frac{n}{3}\right) + 2c \Rightarrow 3^c = n \rightarrow c = \log_3 n$$

$$\Rightarrow T(n) = \log_3 n + 1$$

② آرایه مورد نظر را به ۲ زیر آرایه تبدیل می کنیم و این کار را تا جایی ادامه می دهیم که به تک عضو ها برسیم حال اگر n را با n برابرش مقایسه می کنیم اثر عضو تکراری نبود آنها را یک مجموعه کرده (عضو های تکراری به لول بعد نمی روند) و به لول بعدی می رویم. داریم:

$$A = [1, 4, 8, 16, 32, 64, 128, 256]$$



پایین به بالا دی

عضو میگتا



$$[1, 4, 8, 16, 32, 64, 128, 256]$$

$$[1, 4, 8, 16]$$

$$[32, 64, 128, 256]$$

$$[1, 4]$$

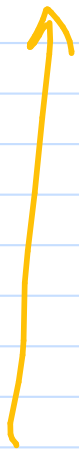
$$[8, 16]$$

$$[32, 64]$$

$$[128, 256]$$

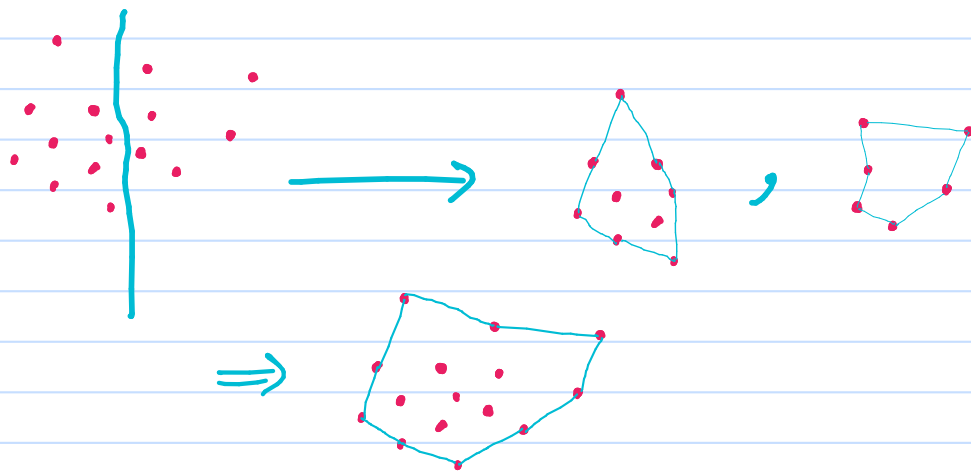


در لایه مرحله از
پایین به بالا مقایسه
می کنیم



۳) ابتدا نقاط را بر حسب x هایشان مرتب می‌کنیم سپس y نقطه یکی با کمترین x و دیگری با بیشترین x را انتخاب می‌کنیم و داریم که تمام نقاط چند ضلعی باید بین این y نقطه باشند، از نقطه‌ی کمینه به نقطه‌ی بیشینه خطی رسم می‌کنیم و در واقع مسئله را به y زیر مسئله تبدیل می‌کنیم با ابعاد کردن همین الگوریتم به صورت بازگشتی چند ضلعی کوچکتری می‌یابیم که شامل همی نقاط درون مجموعه‌اش باشد. در نهایت با بیستی جواب y زیر مسئله را combine کنیم که برای این کار از min, max استفاده می‌کنیم.

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



۴) با توجه به اینکه از روش تقسیم و حل استفاده می‌کنیم ابتدا حالت پایه را مشخص می‌کنیم:

۱) اثر اندازه آرایه‌ای برابر صفر بود، میان آرایه بزرگتر پاسخ است.

۲) اثر اندازه یکی از آرایه‌ها یک شود آنگاه y حالت داریم:

۱- آرایه بزرگتر نیز یک عضو داشته باشد، که در این حالت پاسخ برابر میانگین این

۲ عدد است.

۲- اندازه آرایه بزرگتر از یک بیشتر باشد، که در این حالت با آرایه اول ترکیب کرده و میانه را به کستی آوریم، اگر تعداد زوج بود باید میانگین ۲ عضو وسط را بگیریم اگر تعداد فرد بود عضو وسط یا سقف است (آرایه sorted است).

الگوریتم به این صورت است که: اثر ۲ آرایه a و b داشته باشیم ابتدا $\text{mid}(a)$ و $\text{mid}(b)$ را محاسبه می کنیم حال چند حالت ممکن است رخ بدهد

۱- $\text{mid}(a) = \text{mid}(b)$: با سقف نهایی پیدا شده است که برابر $\text{mid}(a)$ است

۲- $\text{mid}(a) < \text{mid}(b)$: ۲ آرایه را آپدیت کرده به طوری که برای a فقط نیمه راست آن و برای b فقط نیمه چپ را نگه می داریم.

برای یافتن با سقف نهایی الگوریتم را به صورت بازگشتی فرا می خوانیم

۳- $\text{mid}(a) > \text{mid}(b)$: نیمه چپ a و نیمه راست b را نگه می داریم و مجدداً فراخوانی می کنیم.

A) $T(1) = 1 \quad T(2) = 2 \quad T(n) = 1T(n-1) + 1T(n-2)$ (C)

فرض می کنیم $T(n) = O(2^n)$ حال $T(n) \leq C 2^n$ با تجزیه می کنیم

$$T(n) \leq 1C 2^{(n-1)} + 1C 2^{(n-2)} \rightarrow (1+1)C 2^{(n-2)}$$

فرضیه تایید شده و رفتار مجانبی این تابع برابر $O(2^n)$ است

B) $T(1) = 1 \quad T(n) = T(\frac{n}{2}) + T(\frac{n}{2}) + n^2$

فرض $T(n) = O(n^2)$:
 $T(n) \leq C (\frac{n}{2})^2 + C (\frac{n}{2})^2 + n^2 \rightarrow C \frac{n^2}{4} + C \frac{n^2}{4} + n^2$
 تاییدی شود رفتار مجانبی تابع برابر $O(n^2)$ است

$$c) T(1) = 1 \quad T(n) = 4T(\sqrt{n}) + 1$$

فرض می کنیم $T(n) = O(\log n)$

$$T(n) \leq 4c \log(\sqrt{n}) + 1 \rightarrow 4c \log(\sqrt{n} + 1) + 1 \rightarrow 4c \log(n + 1) + 1$$

$$\rightarrow 4c \log n + 4c + 1 \xrightarrow{c = \frac{1}{4}} \log n + 1$$

پس رفتار مجانبی این تابع برابر $O(\log n)$ است

$$d) T(1) = 1 \quad T(2^n) = 7T(2^{n-1}) + 2^n$$

$$T(2^n) = 7T(2^{n-1}) + 2^n \rightarrow T(2^{n-1}) = 7(7T(2^{n-2}) + 2^{n-1}) + 2^n$$

$$\dots \dots \dots \xrightarrow{n \rightarrow 1} 7^n T(2^{n-n}) + (n-1)2^n + 2^n \rightarrow O(7^n)$$

\leftarrow رفتار مجانبی تابع $O(7^n)$