



# ساختمان داده‌ها و الگوریتم‌ها

نیم‌سال اول ۱۳۹۹ - ۱۴۰۰

مدرس: مسعود صدیقین

دانشکده‌ی مهندسی کامپیوتر

## سوالات سری سوم

### مسئله‌ی ۱\*. توابع بازگشتی (۲ نفر)

مرتبه‌ی پیچیدگی هرکدام از توابع زیر را به‌دست آورید.

$$T(n) = \sqrt{2}T(n/2) + \mathcal{O}(\log n) \quad \text{ن}$$

$$T(n) = 2T(\frac{n}{4}) + \mathcal{O}(n) \quad \text{س}$$

$$T(n) = 2T(n-1) + \theta(n) \quad \text{غ}$$

$$T(n) = \sqrt{n}T(\sqrt{n}) + \mathcal{O}(n) \quad \text{ط}$$

$$T(n) = T(\frac{n}{4}) + T(\frac{n}{4}) + T(\frac{n}{8}) + \Omega(n) \quad \text{ض}$$

$$T(n) = T(\lceil \frac{n}{4} \rceil) + 1 \quad \text{ث}$$

$$T(n) = 2T(n/2) + \mathcal{O}(n \log n) \quad \text{خ}$$

### مسئله‌ی ۲\*. ایزی‌ترین سوال

پویا در لابی دانشکده نشسته‌است و مشغول انجام دادن یکی از بی‌شمار تکلیف خودش است. او باید یک تابع بنویسد که دو ماتریس  $n \times n$  را ورودی بگیرد و ماتریس حاصلضرب آن دو را خروجی دهد. او همان الگوریتمی که هنگام محاسبه‌ی این حاصلضرب به صورت دستی انجام می‌دهد را پیاده‌سازی می‌کند و این کد را در سایت سابمیت می‌کند. اما متوجه می‌شود که زمان اجرای آن بسیار طولانی است و مدت زمان بسیار زیادی طول می‌کشد تا سایت نمره‌ی کد او را اعلام کند. بنابراین او تصمیم می‌گیرد الگوریتمی پیدا کند که زمان اجرای این مسئله یعنی ضرب دو ماتریس را کمتر کند.

الف) اگر  $A$  و  $B$  دو ماتریس  $n \times n$  باشند آنگاه ماتریس حاصلضرب آن‌ها  $C = AB$  به صورت زیر محاسبه می‌شود.

$$C_{i,j} = \sum_{k=1}^n A_{i,k} \times B_{k,j}$$

مرتبه‌ی زمانی این الگوریتم را پیدا کنید. فرض کنید ضرب و جمع دو عدد طبیعی از مرتبه‌ی  $\mathcal{O}(1)$  باشد.

ب) حال ایده‌ی اولیه‌ای که پویا به آن رسیده‌است این است که یک ماتریس  $n \times n$  را به صورت یک ماتریس  $2 \times 2$  ببیند که هرکدام از درایه‌های آن یک ماتریس  $\frac{n}{4} \times \frac{n}{4}$  می‌باشند.

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \quad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$$

$$\rightarrow C = \begin{bmatrix} A_{11}B_{11} + A_{12}B_{21} & A_{11}B_{12} + A_{12}B_{22} \\ A_{21}B_{11} + A_{22}B_{21} & A_{21}B_{12} + A_{22}B_{22} \end{bmatrix}$$

نشان دهید پیچیدگی زمانی این حالت با حالت قبل فرقی ندارد!

پ) حال پویا در یک اقدام هوشمندانه ۷ پارامتر را مطابق زیر تعریف می‌کند.

$$p_1 = (A_{11} + A_{22})(B_{11} + B_{22}) \quad , \quad p_2 = (A_{21} + A_{22})B_{11} \quad , \quad p_3 = A_{11}(B_{12} - B_{22})$$

$$p_4 = A_{22}(-B_{11} + B_{21}) \quad , \quad p_5 = (A_{11} + A_{12})B_{22}$$

$$p_6 = (-A_{11} + A_{21})(B_{11} + B_{22}) \quad , \quad p_7 = (A_{12} - A_{22})(B_{21} + B_{22})$$

در این صورت ضرب به صورت زیر خواهد شد.

$$C = AB = \begin{bmatrix} p_1 + p_4 - p_5 + p_7 & p_3 + p_5 \\ p_2 + p_4 & p_1 - p_2 + p_3 + p_6 \end{bmatrix}$$

نشان دهید این حالت مرتبه‌ی زمانی بهتری نسبت به دو حالت قبل دارد.

### مسئله‌ی ۳\*. برش کیک

در این بخش، مساله برش کیک را بررسی می‌کنیم. فرض کنید یک کیک مستطیل شکل داریم که آن را به صورت بازه  $[0, 1]$  مدل می‌کنیم. همچنین  $n$  بازیکن داریم، و هر بازیکن  $p_i$  یک تابع ارزش  $f_i$  دارد که در حقیقت میزان رضایت این بازیکن را از قسمت‌های مختلف کیک نشان می‌دهد. این تابع دارای خاصیت‌های زیر است:

- بازیکن  $p_i$  بازه بین  $a$  تا  $b$  ( $0 \leq a \leq b \leq 1$ ) را به اندازه  $\int_a^b f_i(x) dx$  دوست دارد.

- هر بازیکن کل کیک را به اندازه یک دوست دارد:  $\int_0^1 f_i(x) dx = 1$

- تابع  $f_i$  همه جا مثبت است.

توجه کنید که به ازای دو بازیکن متفاوت  $p_i$  و  $p_j$ ، توابع  $f_i$  و  $f_j$  لزوماً شبیه یکدیگر نیستند. حال، هدف این است که کیک را بین این بازیکن به گونه‌ای تقسیم کنیم که ارزش قطعه‌ای که به هر فرد میرسد، برای او حداقل به میزان  $1/n$  باشد. همچنین فرض کنید که ما از توابع افراد اطلاع نداریم و تنهای می‌توانیم از پرسش‌های زیر جهت به دست آوردن توابع ارزش افراد استفاده کنیم:

- $eval_i(a, b)$ : از بازیکن  $i$  ام سوال می‌کند که بازه بین  $a$  تا  $b$  را به چه اندازه دوست دارد.

- $cut_i(a, v)$ : از بازیکن  $i$  ام می‌خواهد که نقطه  $b$  را مشخص کند، به گونه‌ای که ارزش بازه  $(a, b)$  برای او برابر با  $v$  است.

الف) روشی ارائه دهید که با  $O(n^2)$  پرسش این کار را انجام دهد.

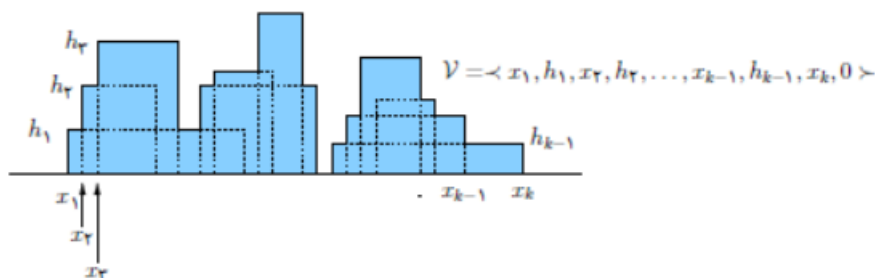
ب) روشی ارائه دهید که با  $(n \log n)$  پرسمان این کار را انجام دهد.

#### مسئله ۴\*. تعداد وارونگی (تقسیم و حل)

در یک آرایه برای  $i < j$  اگر  $A[i] > A[j]$ ، زوج  $i$  و  $j$  یک وارونگی است. الگوریتمی از مرتبه زمانی  $O(n \log n)$  ارائه دهید که تعداد وارونگی‌های یک آرایه را پیدا کند.

#### مسئله ۵. نمای برج‌ها (تقسیم و حل)

$n$  عدد برج بر روی محور  $x$  داریم. هر برج با سه عدد  $l_i$ ،  $r_i$  و  $d_i$  نمایش داده می‌شوند که به ترتیب شروع و پایان برج روی محور  $x$  و ارتفاع آن هستند. می‌خواهیم نمای برج‌ها را که به صورت  $(x_1, h_1, \dots, h_{k-1}, x_k, 0)$  است، به دست آوریم. یعنی از ابتدای محور  $x$  هر جا ارتفاع برج‌ها و در نتیجه نما تغییر می‌کند و مختصات آن نقطه و ارتفاع مذکور را می‌خواهیم. الگوریتمی از مرتبه زمانی  $O(n \log n)$  ارائه دهید.



#### مسئله ۶\*. نزدیکترین جفت (تقسیم و حل)

$n$  نقطه در فضای دو بعدی داده شده است. الگوریتمی ارائه دهید که در زمان  $O(n \log n)$  نزدیکترین جفت این نقاط را پیدا کند.