ساختمان داده‌ها و الگوریتم‌ها

نیم‌سال دوم ۹۷ــ۹۸

دانشکده‌ی مهندسی کامپیوتر

گردآورندگان: عرفان فرهانی، محمدرضا احمدخانی

۶ اسفند | پاسخ کوئیز نخست | زمان: ۳۰ دقیقه

مسئله‌ی ۱. تحلیل پیچیدگی

پیچیدگی توابع زیر را تحلیل کنید

الف)

$$f(n) = lg(n!)$$

حل.

$log(n!) = \sum_{i=۱}^{n} log(i)$

$\sum_{i=۱}^{n} log(i) \leqslant \sum_{i=۱}^{n} log(n) = nlog(n)$

$\sum_{i=۱}^{n} log(i) \geqslant \sum_{i=\frac{n}{۲}}^{n} log(i) \geqslant \sum_{i=\frac{n}{۲}}^{n} log(\frac{n}{۲}) = \frac{n}{۲}log(\frac{n}{۲})$

$\Rightarrow lg(n!) \in \Theta(nlgn)$

$\triangleright$

ب)

$$T(n) = ۳T(\frac{n}{۳}) + \sqrt{n}$$

حل.

با استفاده از حالت اول قضیهٔ اصلی اثبات می‌شود که:

$$\sqrt{n} \in O(n^{log_۳ ۳ - \epsilon}) \Rightarrow T(n) \in \Theta(n^{log_۳ ۳}) = \Theta(n)$$

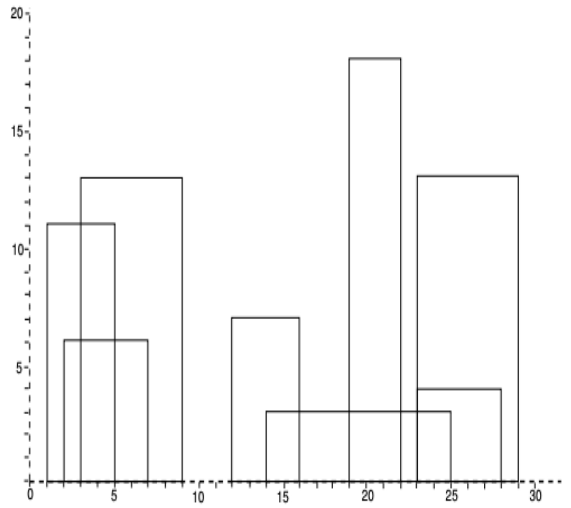
$\triangleright$

مسئله‌ی ۲. skyline

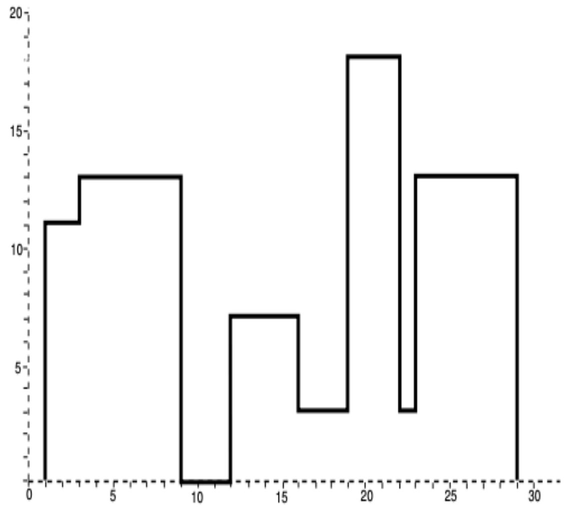
n ساختمان مستطیل شکل در یک شهر دو بعدی داده شده اند. هر ساختمان با (Li،Hi،Ri) مشخص می‌شود که به ترتیب از چپ به راست نشان دهنده‌ی مختصات شروع ساختمان(سمت

چپ)، ارتفاع آن و مختصات انتهای ساختمان(سمت راست) است. اگر مستقیم به این ساختمان‌ها نگاه کنیم برخی خطوط قابل رویت نیستند. با پاک کردن آن خطوط، اسکایلاین ساختمان‌ها به دست می آید. به عبارت دیگر اسکایلاین برابر است با ناحیه ای که توسط این مستطیل‌ها اشغال شده است (اجتماع آن‌ها).
الگوریتمی پیشنهاد دهید که اسکایلاین n ساختمان را در زمان $O(nlogn)$ به دست آورد.

مثال:
در ابتدا مشخصات ساختمان ها داده می شود:
{(۳, ۱۳, ۹), (۱, ۱۱, ۵), (۱۲, ۷, ۱۶), (۱۴, ۳, ۲۵), (۱۹, ۱۸, ۲۲), (۲, ۶, ۷), (۲۳, ۱۳, ۲۹), (۲۳, ۴, ۲۸)}
جواب، اسکایلاین به صورت یک سری مختصات x است که با ارتفاع های مختلف به هم وصل شده اند.(از چپ به راست)



buildings



skyline

$skyline$: {۱, **۱۱**, ۳, **۱۳**, ۹, **۰**, ۱۲, **۷**, ۱۶, **۳**, ۱۹, **۱۸**, ۲۲, **۳**, ۲۳, **۱۳**, ۲۹, **۰**}

اسکایلاین به ترتیب x ها است.

از $x = ۱$ تا $x = ۳$ با ارتفاع ۱۱، از $x = ۳$ تا $x = ۹$ با ارتفاع ۱۳ و...

حل.

برای حل این سوال از تقسیم و حل استفاده میکنیم. ایده‌ی کلی بسیار شبیه به مرتب‌سازی ادغامی است.
اگر $n = ۱$ باشد آن گاه اسکایلاین، خود ساختمان می‌شود.

در هر گام، ساختمان‌ها را به دو قسمت تقسیم میکنیم. پس از به دست آوردن اسکایلاین مربوط به هر قسمت، آن دو را به صورت زیر ادغام می‌کنیم:

۱. هر اسکایلاین شامل یک سری مختصات x و مختصات y (ارتفاع ساختمان) می‌شود.

چپ‌ترین مختصات دو اسکایلاین را در نظر می‌گیریم.

۲. نقطه با مختصات x کمتر را انتخاب می‌کنیم.

۳. ارتفاع این نقطه را با ارتفاع نقطه‌ی قبلیِ انتخاب‌شده در اسکایلاین دیگر مقایسه می‌کنیم (اگر نقطه‌ي موجود در اسکایلاین دیگر، اولین نقطه باشد، ارتفاع نقطه‌ی قبلی را صفر میگیریم) اگر ارتفاع نقطه‌ی انتخاب‌شده کمتر از نقطه‌ی دیگر باشد، ارتفاعش را به ارتفاع نقطه‌ي دیگر افزایش می‌دهیم (به عبارتی این نقطه، در اسکایلاین ادغام‌شده محو شده است) و در غیر این صورت کاری انجام نمی‌دهیم.

۴. نقطه‌ی بعدی در اسکایلاین انتخاب شده را در نظر می‌گیریم.

۵. تا زمانی که نقاط بررسی نشده‌ای موجود باشند مراحل ۲ تا ۴ را انجام می‌دهیم.

۶. نقاط اضافی (نقاط پشت سر همی که شانx متفاوت است ولی ارتفاعشان برابر است) را پاک می‌کنیم.

این ادغام در زمان خطی قابل انجام است (چرا؟) پس در کل داریم:

$T(n) = ۲T(n/۲) + o(n)$

و در نتیجه:

$T(n) = nlog(n)$

هم‌چنین در ابتدا تنها پیش‌پردازشی از زمان $O(nlogn)$ برای مرتب کردن ساختمان‌ها بر اساس مختصات x شان لازم داریم.

$\triangleright$

مسئله‌ی ۳. قضیه اصلی

آیا می‌توان قضیه اصلی را برای تابع زیر استفاده کرد؟ چرا؟

$T(n) = ۴T(\frac{n}{۲}) + n^۲lgn$

تابعی مانند $f(n)$ ارائه دهید که $T(n) \in O(f(n))$

حل.

به دلیل آن‌که $f(n) = n^۲lgn \notin O(n^{log_۲ ۴-\epsilon})$ و $f(n) \notin \Omega(n^{log_۲ ۴+\epsilon})$ نمی‌توانیم از قضیهٔ اصلی استفاده کنیم. حدس می‌زنیم که $T(n) \leqslant cn^۲lg^۲n$

$$T(n) \leqslant ۴T(n/۲) + n^۲lgn = ۴c(n/۲)^۲lg^۲(n/۲) + n^۲lgn$$

$$= cn^۲lg(n/۲)lgn - cn^۲lg(n/۲)lg۲ + n^۲lgn$$

$$= cn^۲lg^۲n - cn^۲lgnlg۲ - cn^۲lg(n/۲) + n^۲lgn$$

$$= cn^۲lg^۲n + (۱ - c)n^۲lgn - cn^۲lg(n/۲) \quad (c > ۱)$$

$$\leqslant cn^۲lg^۲n - cn^۲lg(n/۲) \leqslant cn^۲lg^۲n$$

$\triangleright$

موفق باشید :)